university of toronto
computing services

Computer News

SEASON'S
GREETINGS

ISSN 0315-4661 November/December 1988 No. 251

Contents

HyperCard: Should you be interested 1
KanjiCard Takes Off 8
The History of KanjiCard 9
BIB: A Bibliographic Application of HyperCard 11
Early HyperCard Stacks 13
Subscription Renewals 14
Return of Domesday and Much More 14
Macintosh Virus Prevention 16
ARCHIVIST: Database Manager and Text Processor 18
COMMODORE PC 60-40 20
Microcomputer Courses for the New Year 23
Site License for SYSAT on Macs 24
Numerical and Scientific Software for Microcomputers 25
Computer Algebra: Symbolic Manipulations Software 27
SPSS^x Version 3.1 29
GAUSS News 30
WordPerfect 4.2 Course to be Phased Out 30
Minpack Library Available on CMS 31
MINOS Available on CMS 32
WordPerfect 1.0 for the Macintosh 33
VM/CMS Lanugage Processor Upgrade 36

Departments

Documents at UTCS 37
Consulting and Enquiries 38
UTCS Directory 38
Committees on Computing 38
UTCS Terminal and Advising Sites 40
UTCS Services 41
UTCS Software List 42

Cover: by E. Weitmann

Season's Greetings from all of us at UTCS.

Acknowledgements: Many thanks to Mark Vanderkam for assisting in the editing of this issue.

Publisher
University of Toronto Computing Services
Toronto, Ontario
Canada M5S 1A1

Editor: Martha Parrott

Publications Assistant: Elizabeth Weitmann

Publication Date: *December 1, 1988*

Subscriptions

Subscriptions run for one year, beginning each January. Changes should be entered on the form at the back of this issue and mailed to *ComputerNews*, 4 Bancroft Avenue, Room 217 (or phone 416-978-4034).

ComputerNews is published eight times yearly by the University of Toronto Computing Services (UTCS). Enquiries should be directed to:

The Editor
ComputerNews
University of Toronto Computing Services
4 Bancroft Avenue, Room 217
Toronto, Ontario
Canada M5S 1A1

Network address:
PARROTT at UTORONTO
(NetNorth/BITNET/EARN)

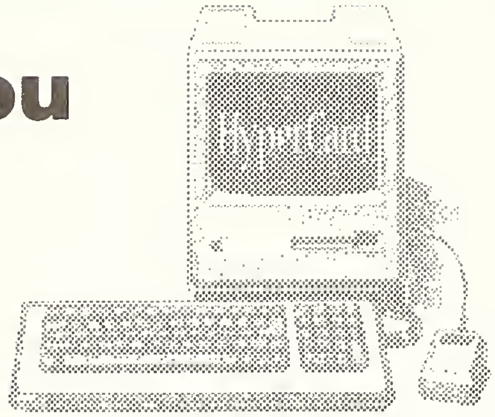
Unless otherwise indicated, network addresses appearing after the author's name pertain to NetNorth/BITNET/EARN.

Permission is granted to reprint articles from *ComputerNews* for noncommercial purposes provided the author, publication, and issue are acknowledged.

IBM PC, PC-XT, PC-AT, Personal System/2, Operating System/2 are trademarks of IBM.
LaserWriter is a registered trademark of Apple Computer.
Lotus 1-2-3 is a registered trademark of Lotus Corporation.
Macintosh 512K, Plus, SE are trademarks of Apple Computer.
Microsoft Word is a trademark of Microsoft Corporation.
PostScript is a registered trademark of Adobe Systems Inc.
SAS is a trademark of SAS Institute Inc., Cary, N.C.
SPSS^x is a trademark of SPSS Inc.
UNIX is a trademark of AT & T Laboratories.

This newsletter was entirely produced on a Macintosh using PageMaker and other related Macintosh software.

HyperCard: Should you be interested?



JohnBradley
bradley@vm.utcs.utoronto.ca

In August 1987, Apple released a new piece of software called HyperCard. Written by Apple "whiz kid" Bill Atkinson, and announced by Apple with much fanfare, it has continued to generate significant interest and curiosity. In the preface to Danny Goodman's book *The Complete HyperCard Handbook*, Apple chairman and CEO John Sculley called Atkinson "a national treasure" in "the land of Apple Computer." Indeed, Apple has made so much of HyperCard that many people think it is appropriately named: a product that has received a lot of "hype."

Should you be interested in HyperCard? Well, UTCS believes that certain types of activities at U of T could potentially benefit from it. This issue of *ComputerNews* presents several articles on HyperCard, in which we hope to give you a sense of what it is all about.

A little history

One of HyperCard's origins, and the real origin of its name, is found in the field of Hypertext, whose history begins in 1945. In that year, Vannevar Bush, the science advisor to Franklin D. Roosevelt, wrote an article for *The Atlantic Monthly* called "As We may Think." In that article, Bush recognized the rapid growth of information in the world and concluded that a time would come when most people might access that information principally by a machine. He proposed a machine called a "memex," a device in which an individual would store all his or her books, records, and communications. The memex owner could consult a book by entering a code for the book, which would then be displayed on a screen. The viewer could add personal comments or marginal notes to this material. S/he could view more than one book at the same time and could build associations between material in the various books on view. The associations would be saved by the memex. Thereafter, anytime either reference was accessed, the other could be referenced as well, merely by pushing a button. These associations acquired the name "links" and formed the basis of the idea of Hypertext.

Bush's ideas were impossible to carry out in his day; it was not until the 1960s that it became possible to implement even some of his ideas. At that time Ted Nelson and Doug Engelbart began to design computer systems that would allow their users to manipulate text and create links between material in different documents. Engelbart and Nelson viewed traditional text files as two-dimensional objects that presented material in an essentially *flat* manner. In their view, the linking of points in a conventional flat docu-

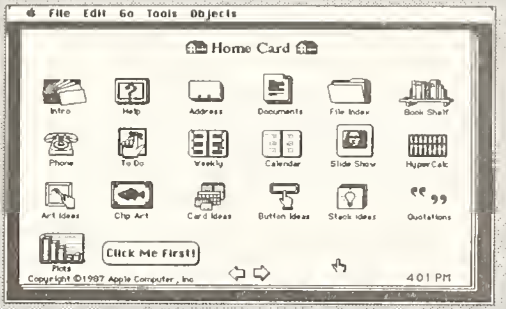
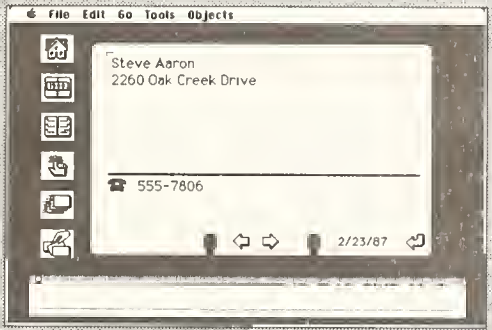


Figure 1

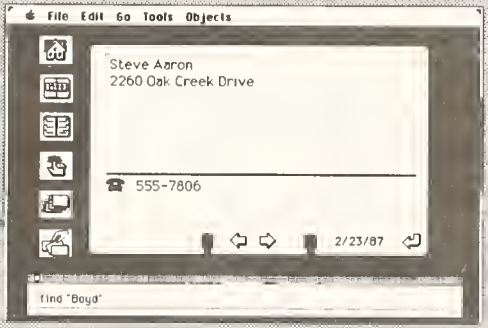


Steve Aaron
2260 Oak Creek Drive

☎ 555-7806

2/23/87

Figure 2



Steve Aaron
2260 Oak Creek Drive

☎ 555-7806

2/23/87

find "Boyd"

Figure 3

ment, or between two points in different documents, added another dimension to a collection of texts: hence the term *hypertext*.

Until the advent of the personal computer and the development of software such as Atkinson's MacPaint, computers could handle graphical images only clumsily, if at all. When graphics became as convenient to create as text, people began to extend the idea of hypertext to *hypermedia*. The merging of some of the ideas of Nelson's hypertextual systems with Atkinson's own ideas about how to work with graphics on the Macintosh resulted in HyperCard. The debate still rages among hypertext experts whether enough of the hypertext ideas of Nelson and Engelbart are present in HyperCard for it to qualify as a true hypertext system. HyperCard users, however, have found the mixture to result in a powerful and versatile development tool.

As we said earlier, Apple announced the first version of HyperCard in the summer of 1987. Any Apple Macintosh computer sold after that date included a free copy of HyperCard. You can also purchase it separately for older Macintoshes. The retail price in Canada is $78, but undoubtedly some Apple dealers sell it for significantly less. By making HyperCard available in this form, Apple has ensured that it is widely available. Hence, although it is only one of several hypermedia systems available, it is the only one most computer users have heard of. Although HyperCard focuses on textual and graphical media, it is possible to extend HyperCard in various ways to include animation or high quality sound.

What is HyperCard?

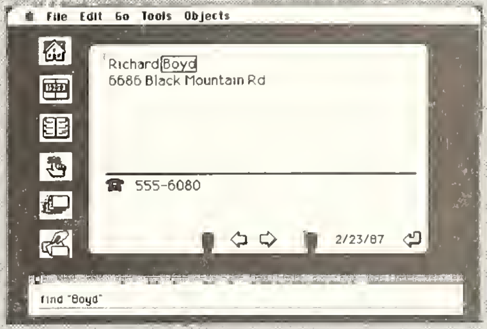
"HyperCard is not an application, like a word processing, spreadsheet, or database program. It is, rather, a software engine — a hypermedia toolkit, if you will." (John Sculley)

Although HyperCard has been available for a little over a year, there is still considerable discussion over its exact nature and purpose. This article presents a brief overview of HyperCard; it is followed by others illustrating some uses of HyperCard.

When you install HyperCard on your Macintosh, you can use two parts of the same product:

- The basic HyperCard program — Sculley's "software engine."
- A collection of HyperCard *stacks*. These are HyperCard applications. When you click on a HyperCard stack, you start the HyperCard program, which uses the information in the stack to tell it what do to.

All HyperCard applications use a model of one or more collections of related *cards* called stacks. At any point while the application is running, your screen shows a single card from one of your stacks. Each card can contain text and graphics. In addition, certain areas of any card can be set up as *buttons*. Usually, if you use the Macintosh mouse to click on a button, an action occurs that is associated with that button. Hence a stack is not only the *text and graphics* on the individual cards, but also the *set of actions* associated with the various objects (usually buttons) on the cards. Computer



Richard Boyd
6686 Black Mountain Rd
☎ 555-6080

find "Boyd"

Figure 4

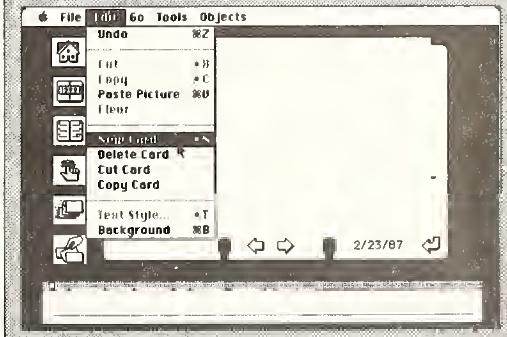


Figure 5

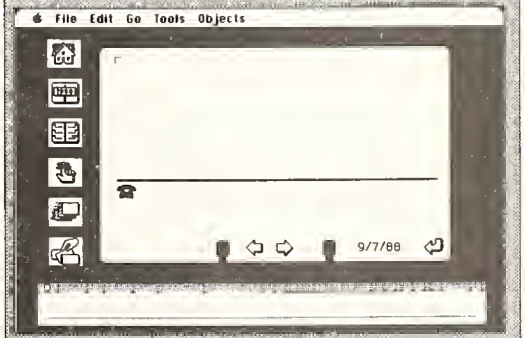


Figure 6

applications that are built as stacks are usually called "stackware."

A simple example is the HyperCard "Home" stack. For many people, the Home Stack is the main entry point into HyperCard. If you click on the Home Stack icon, you see something similar to *Figure 1*.

The screen shows the first card (called the Home Card) in the Home Stack. In the Home Card, the mouse pointer appears as a hand with one finger extended, ready to push a button. You can see the mouse pointer near the bottom right corner of *Figure 1*. On the card, you see pictures representing the various other basic stacks that Apple provides with HyperCard. The image is simply a picture, similar to what you could draw with MacPaint. Also associated with the card, but invisible to you, is a collection of buttons. Associated with each icon (such as the little picture called "Intro", or "Help") is a button that, when pushed, makes HyperCard take you to the stack of that name.

How does HyperCard know what to do when you push a button? HyperCard comes with a language called HyperTalk, which allows you to say, for example, that when a particular button on a card is pushed, HyperCard should jump to a particular card in this or another stack. Although HyperTalk is not directly visible to beginning HyperCard users, it is always behind the scenes, telling HyperCard what to do. HyperCard users can easily examine and change any of these HyperTalk instructions.

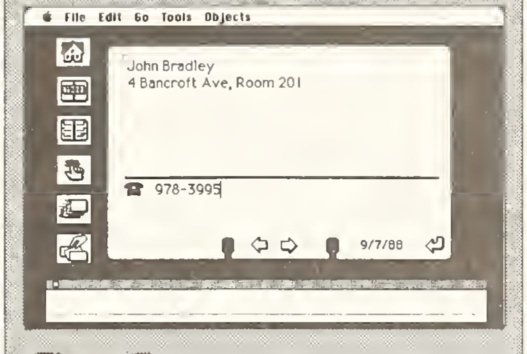
Essentially, you see with the Home Card all the basic components of the HyperCard system:

- A basic organizational system that allows you to structure material into cards and collections of cards called stacks.
- A graphical presentation system that allows you to present material through text and graphics, in a manner similar to MacPaint.
- A language called HyperTalk that tells HyperCard what action to take when a stack user does something, such as click on a particular area on a card.

In addition, HyperCard can be extended through the use of External Commands, programs that can be invoked from within HyperCard stacks and that provide functions not normally available from HyperCard itself. For example, we mentioned earlier that HyperCard could be extended to do graphics and high quality sound. It is through these External Commands that such features are usually added.

How can *you* use HyperCard?

Apple claims that HyperCard gives everyone the tools to become a software author, to create an information-based application that looks professionally designed. They believe that HyperCard gives average Macintosh users the ability to create new things on the Macintosh in the same way that BASIC gives IBM PC users the ability to create new programs on the PC. Like BASIC,



John Bradley
4 Bancroft Ave, Room 201

☎ 978-3995

9/7/88

Figure 7

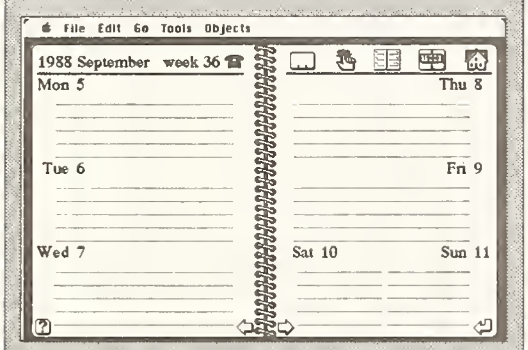
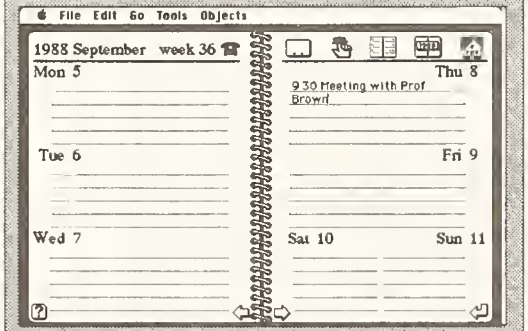


Figure 8



9 30 Meeting with Prof Brown

Figure 9

HyperCard hides a considerable amount of complexity that you would encounter if you tried to do the same thing with other programming languages.

Although HyperCard allows you to create almost any kind of computer program, people commonly use HyperCard to do three types of things:

1. **Navigation**: The stack author creates a stack already containing the information to be conveyed to you, the stack user. HyperCard is used to present you with various types of information, and it allows you to manuever within the stack. In most Navigation applications, the stack author puts all or most of the information into the stack, and you only use HyperCard to examine it. A recent prominent Navigational stack was one written for visitors to Toronto's June Summit. It was constructed to provide some information about Toronto and included sections on public transit, restaurants, and places of interest.

2. **Database**: You use HyperCard to store small databases such as a telephone list, or a calendar of activities. In Database applications, the stack author creates a set of tools within the stack to make it easy for you to store and access your own information.

3. **Prototyping**: HyperCard is used by a computer programmer to construct a preliminary version of an application program so that people can try it out before investing the effort and expense to build the real thing. Here the parallel with BASIC is useful to consider again. HyperCard is used to build the prototype because it is easy to do relatively sophisticated programming-like tasks in HyperTalk; however, the application itself may then be translated into a more efficient standard programming language

Some sample HyperCard stacks

A great deal of free stackware is available. Apple gives you some sample stacks that could be useful in their own right. In HyperCard you can modify nearly any stack to suit your own needs. By looking at a few of these readily available stacks, you will better understand what HyperCard could be used for.

HyperCard's Personal Organizer Stacks

Let's take a brief look at some of the stacks that Apple gives you. Many of the stacks they give you work together to provide a basic Personal Organizer system. We'll look first at the Address stack, by clicking on the Address button on the Home Stack.

In the Address stack is a collection of cards similar to those shown in *Figure 2*. To the left is a row of icons, each representing a different

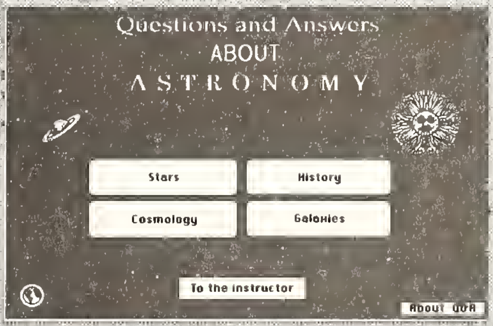
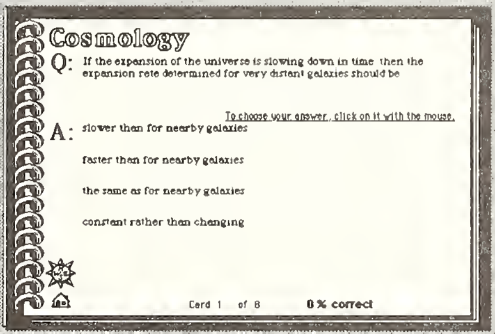


Figure 10



Cosmology

Q: If the expansion of the universe is slowing down in time then the expansion rate determined for very distant galaxies should be

To choose your answer, click on it with the mouse.

A: slower than for nearby galaxies

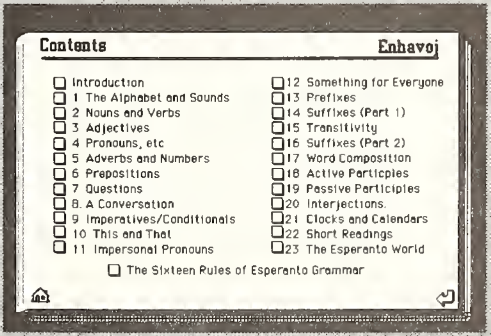
faster than for nearby galaxies

the same as for nearby galaxies

constant rather than changing

Card 1 of 8 0 % correct

Figure 11



Contents Enhavoj

☐ Introduction
☐ 1 The Alphabet and Sounds
☐ 2 Nouns and Verbs
☐ 3 Adjectives
☐ 4 Pronouns, etc
☐ 5 Adverbs and Numbers
☐ 6 Prepositions
☐ 7 Questions
☐ 8. A Conversation
☐ 9 Imperatives/Conditionals
☐ 10 This and That
☐ 11 Impersonal Pronouns
☐ 12 Something for Everyone
☐ 13 Prefixes
☐ 14 Suffixes (Part 1)
☐ 15 Transitivity
☐ 16 Suffixes (Part 2)
☐ 17 Word Composition
☐ 18 Active Participles
☐ 19 Passive Participles
☐ 20 Interjections.
☐ 21 Clocks and Calendars
☐ 22 Short Readings
☐ 23 The Esperanto World
☐ The Sixteen Rules of Esperanto Grammar

Figure 12

stack. If you click on the house icon, for example, HyperCard returns you to the Home Stack (as shown in *Figure 1*). The main part of the screen, to the right of the icons, shows a particular card with the name, address, and telephone number of a Mr. Steve Aaron. There is one card for each address in the stack. You can look at other addresses in the stack by clicking on the left and right arrows at the bottom of the card. These arrows are connected to buttons that move you forward or backward through your cards to other addresses.

Let's say that you want to look up the address for a Mr. Boyd, who you know is in the stack. For this we can use the Message Box, shown at the bottom of the screen. To do that, we type a HyperTalk command **find "Boyd"** in the box (*Figure 3*). HyperCard looks through all the cards to find one containing the text "Boyd". Almost immediately, it finds the appropriate card and presents it to you (*Figure 4*).

The Address stack is a Database type of application, discussed above. To add a new card for a new address, we first choose the option New Card from the Menu Bar (*Figure 5*). HyperCard then inserts a blank card ready for you to type in a new name and address (*Figure 6*). By typing in a new name and address (*Figure 7*), you are adding both a new card and new data to the stack.

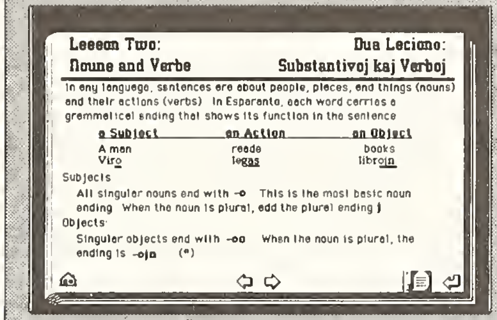
So far we haven't really seen much of the linking capabilities of HyperCard. To demonstrate a link to another stack, we could click on the weekly calendar icon (shown as a small weekly calendar, the third icon from the top in *Figure 2*). Overlaying this icon on the card is a button that, when pushed, causes HyperCard to link to the "Weekly" calendar stack and move to the card containing information about the current week. The resulting screen is similar to that shown in *Figure 8*.

Now we've used a HyperCard link to change from the Address stack to the Weekly calendar stack. We can see from *Figure 8* that the card contains no personal data. Suppose we want to add an appointment to Thursday the eighth. We simply place the cursor on the line under Thursday the eighth and type, with a result something like *Figure 9*. We have now added new material to this card; anytime we look at this card in the future, this information will be there.

If you own HyperCard, you are not restricted to using the stacks that Apple provides; you can produce your own. Because of this, many people have begun to produce both shareware and commercial products based on HyperCard. Here is a brief look at a few of these:

A simple CAI application

Computer Aided Instruction (CAI) may become a common use of HyperCard in the future. *Figure 10* shows the first card of a shareware stack written by Douglas Wood (© 1987), which presents a simple quiz on astronomy called Q&A. As



Lesson Two: **Dua Leciono:**
Nouns and Verbs **Substantivoj kaj Verboj**

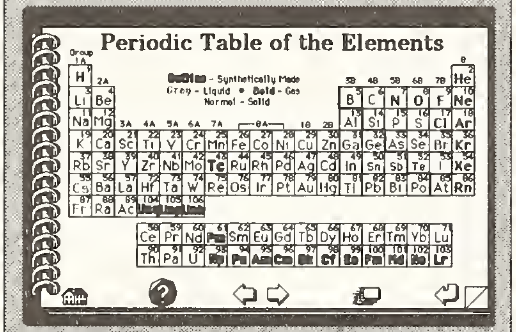
In any language, sentences are about people, places, and things (nouns) and their actions (verbs). In Esperanto, each word carries a grammatical ending that shows its function in the sentence.

a Subject	an Action	an Object
A man	reads	books
Viro	legas	librojn

Subjects:
All singular nouns end with **-o**. This is the most basic noun ending. When the noun is plural, add the plural ending **j**.

Objects:
Singular objects end with **-on**. When the noun is plural, the ending is **-ojn** (*)

Figure 13

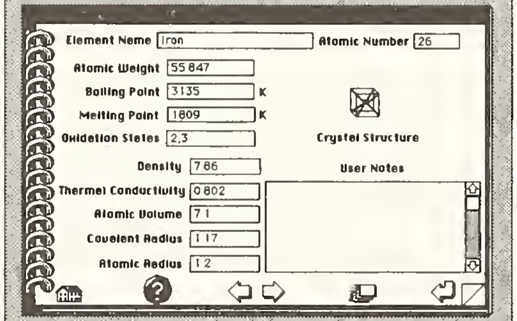


Periodic Table of the Elements

[illegible] - Synthetically Made
Grey - Liquid • **Bold** - Gas
Normal - Solid

Group 1A	2A	3A	4A	5A	6A	7A	8A			1B	2B	3B	4B	5B	6B	7B	8
1 H																	2 He
3 Li	4 Be											5 B	6 C	7 N	8 O	9 F	10 Ne
11 Na	12 Mg											13 Al	14 Si	15 P	16 S	17 Cl	18 Ar
19 K	20 Ca	21 Sc	22 Ti	23 V	24 Cr	25 Mn	26 Fe	27 Co	28 Ni	29 Cu	30 Zn	31 Ga	32 Ge	33 As	34 Se	35 Br	36 Kr
37 Rb	38 Sr	39 Y	40 Zr	41 Nb	42 Mo	43 Tc	44 Ru	45 Rh	46 Pd	47 Ag	48 Cd	49 In	50 Sn	51 Sb	52 Te	53 I	54 Xe
55 Cs	56 Ba	57 La	72 Hf	73 Ta	74 W	75 Re	76 Os	77 Ir	78 Pt	79 Au	80 Hg	81 Tl	82 Pb	83 Bi	84 Po	85 At	86 Rn
87 Fr	88 Ra	89 Ac	104 [illegible]	105 [illegible]	106 [illegible]												
			58 Ce	59 Pr	60 Nd	61 Pm	62 Sm	63 Eu	64 Gd	65 Tb	66 Dy	67 Ho	68 Er	69 Tm	70 Yb	71 Lu	
			90 Th	91 Pa	92 U	93 Np	94 Pu	95 Am	96 Cm	97 Bk	98 Cf	99 Es	100 Fm	101 Md	102 No	103 Lr	

Figure 14



Element Name: Iron
Atomic Number: 26
Atomic Weight: 55.847
Boiling Point: 3135 K
Melting Point: 1809 K
Oxidation States: 2,3
Density: 7.86
Thermal Conductivity: 0.802
Atomic Volume: 7.1
Covalent Radius: 1.17
Atomic Radius: 1.2

Figure 15

Douglas Wood describes it:

> Q&A is a general purpose "question and answer" drill facility for education. Students are presented with a question and 5 possible answers. [See *Figure 11* for an example.] They select the answer of their choice by clicking on it. A saxophone will play if they got the right answer, a boing if they did not. The program will keep a running percentage of their correct answers. When they reach the last card in the stack they will be put into a "review" mode which will go back through the questions that they missed giving them a chance to try to answer the question again. In review mode they will have a chance to find out why the correct answer is the one given by clicking on the ? icon. Q&A is not intended to be a testing tool; rather its purpose is to provide students with a non-competitive chance to test their knowledge of a subject.

In the Q&A stack, HyperCard is used as a navigational tool — HyperCard leads the student through an exercise. The student doesn't store any new material in the stack but simply views material put there by the stack creator. By copying the stack and changing the questions and answers in the new stack, the instructor can build an exercise that uses the same model.

A HyperCard Esperanto Stack

Another CAI-oriented stack is the Esperanto stack, written by Michael Urban. The card that shows the contents of the stack is shown in *Figure 12*. If we click on the box beside item 2, HyperCard moves to the card shown in *Figure 13*. Of particular interest is the "speaking face" logo shown in the bottom right corner of this card. Attached to this spot is a button. If you click on this button, you can type a letter combination, and HyperCard invokes an external command that plays the corresponding Esperanto pronunciation through the Macintosh's speaker.

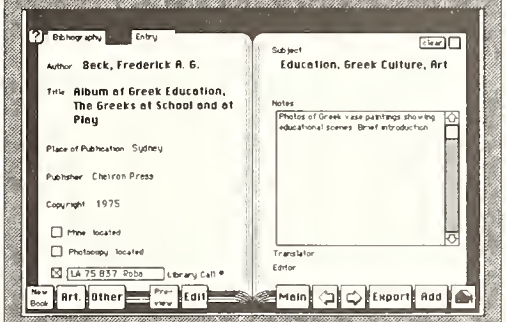
A HyperCard Periodic Table Stack

The Periodic Table, written byAndrew D. Jewell, is yet another navigational stack. When this stack is started, you see the Periodic Table displayed (*Figure 14*). Additional information about each element is stored in the stack. To view it, you simply click on the element you want: the stack writer has put a button over each element entry, which takes you to the appropriate card. If we click on "Fe", for example, we see the "Iron" card (*Figure 15*).

Jewell has also included a field called "User Notes" where you can type in some information you want to save about this element. As with most other HyperCard stacks, you can change the definition of a stack as you wish, adding and removing information, buttons, and links to other stacks or cards.

A HyperCard Bibliography Stack

As a final example, we'll take a brief



Author Beck, Frederick A. G.
Title Album of Greek Education, The Greeks at School and at Play
Place of Publication Sydney
Publisher Cheiron Press
Copyright 1975
☐ Mine located
☐ Photocopy located
☒ LA 75 B37 Roba Library Call #
Subject Education, Greek Culture, Art
Notes Photos of Greek vase paintings showing educational scenes. Brief introduction
Translator
Editor

Figure 16

look at a prototype stack created here at U of T by Geoffrey Rockwell, one of the people in the Apple Research Partnership Program (ARPP) here at U of T. This stack is used to store bibliographical information. Each card contains information about a particular book or article (see a sample in *Figure 16*, and you can easily add new cards and type in new information.

When you have created your database you can use the stack to sort it or find a particular card (for example, you could ask HyperCard to find all books by Frederick Beck, or find all cards that contain "Greek Culture" in the Subject field). You can export a list of selected items as a bibliography that could be included in an article or report you are writing.

For more information on this stack, see "BIB: A Bibliography in HyperCard" on page 11.

HyperCard's future

Last April, Apple announced a new version of HyperCard that can work with CD-ROM. CD-ROM is a computer data storage system built on the same technology used for compact discs. Instead of recording music on the disc, computer data is put there.

CD-ROMs hold a large amount of data (about 500 megabytes). Since CD-ROM discs are produced in the same way as compact discs and cannot be generated directly by a personal computer, most people expect CD-ROMs to be a publishing medium for data, rather than a personal storage medium. Apple has viewed HyperCard as a natural engine to navigate through the large amounts of data that could be published on a CD-ROM and has released a demonstration disk that uses HyperCard in this way. The demonstration disk contains several sample stacks, including excerpts from:

- The *Whole Earth Catalog.*
- An interactive form of *Grolier's Encyclopaedia*, called "The California Gold Rush."
- The Electric Cadaver, a stack that allows you to examine various parts of the human anatomy (see article on page 13).
- The Perseus Project, a project that introduces various aspects of classical Greek culture and history.

These stacks are not simply in the electronic form of a book. They all take advantage of the flexibility of HyperCard and show a more dynamic method of presentation. Apple is clearly banking on various publishing companies offering material in a CD-ROM format. Currently, although the demonstration disk suggests that work is proceeding on at least these products, so far none of them is available commercially. Only time will tell whether a significant number of products in CD-ROM format for the Macintosh actually materializes.

Will HyperCard ever be available on the IBM PC? Obviously, Apple owns the product and is unlikely to make an equivalent version to run on the competition. However, there is some interest in somewhat similar products for the PC. For example, Owl International markets a product called Guide, which is available for both the IBM PC and the Macintosh. If HyperCard is successful, something like it will probably appear for the IBM computers. However, since HyperCard relies heavily on the Macintosh's famous user interface, you cannot expect a very similar product for the PC until a similar interface is widely available and well established. (Indeed, Owl's Guide program runs under Microsoft Windows.)

Apple is betting that in a year or two most Macintosh owners will have HyperCard. If this happens, products written to run under HyperCard will have a large potential market ready made. The situation is not so clear for the IBM PC. Although OS/2 and the Presentation Manager (for the IBM AT or PS/2) will eventually offer a similar environment, it looks like OS/2 and the Presentation Manager will not be widely available on IBM Personal Computers for at least a year or so. Most IBM PCs or compatibles on the U of T campus today will not be able to run OS/2.

What can UTCS do for you?

UTCS has been developing expertise in HyperCard for about a year now. We feel that we can:

- Help you learn more about what HyperCard is about, to see if you could make use of it.
- Answer your questions about using HyperCard.

In addition, we are preparing an introductory HyperCard seminar. Apple's ARPP people at U of T already offer an introductory course for those those who wish to develop their own HyperCard stacks.

Please give us a call at 978-HELP, if you would like more information.

KanjiCard Takes Off

Mark Vanderkam
pubscoop@vm.utcs.utoronto.ca

The April *ComputerNews* cover story told of an impressive new computer application that helps people learn Japanese. KanjiCard, which in April was a prototype, has changed and developed so much in the last six months that it warrants another article.

KanjiCard uses a comprehensive array of graphics, animations, sound recordings, and pages of text to help users memorize Kanji, the Chinese characters used in written Japanese. KanjiCard's creators believe that a computer presentation can familiarize students with the different aspects of the Kanji so that students can learn the characters faster and more thoroughly. People who saw the prototype were impressed that it could demonstrate how to write Kanji characters, how to voice them, how to read them with other Kanji, and how to interpret their meaning. Even last spring, KanjiCard seemed destined to become a hit.

After the prototype was approved for production, five people were hired in April to work full-time on KanjiCard. A demonstration version was shown at conferences and to language instructors and students, and the resulting suggestions led to five structural revisions. As the work progressed, commercial vendors began to show an interest. Apple Canada, which has supported this work from the beginning (KanjiCard was created and can only be used on Apple's Macintosh computers), agreed to press the initial KanjiCard package on compact discs. These discs will be distributed by the Computer Writing Network Project in East Asian Languages (which oversees KanjiCard) as an experimental package.

This initial package will contain 50 out of a projected 300 Kanji characters. The developers feel that an early release is necessary to allow language professionals to begin using and evaluating KanjiCard. Once people start using it, KanjiCard may be revised or reformatted. Over years of study, language students learn a basic set of 750 Kanji characters. To create a system containing these may require a new software and hardware format. Interactive videodiscs may be needed to store video images and other forms of information.

While it is impossible to tell how KanjiCard will change in the coming year, we have noted some new features of KanjiCard, and some problems encountered while developing the application. See the next article for a look at the history of KanjiCard's development.

The History of KanjiCard

Mark Vanderkam
pubscoop@vm.utcs.utoronto.ca

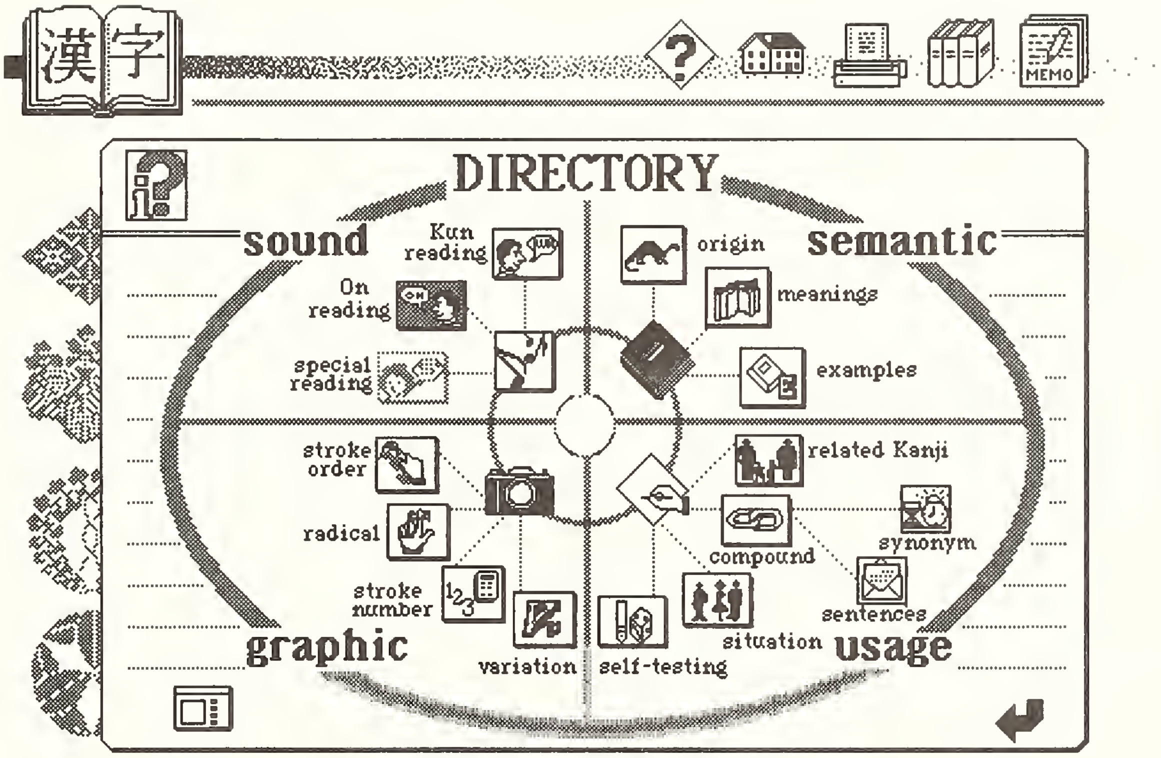
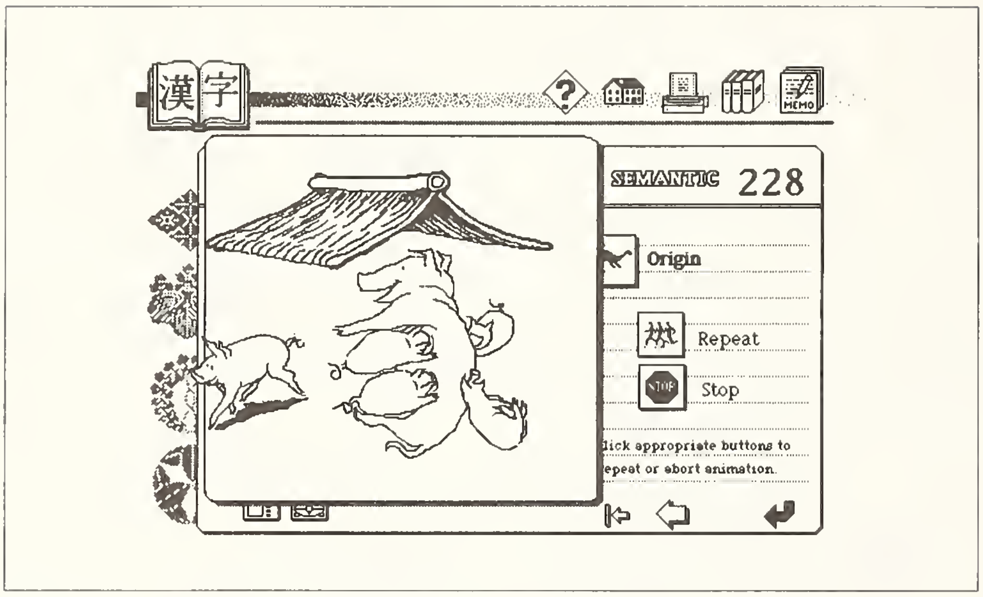


Figure 1: KanjiCard Directory

One of the many interesting things about KanjiCard is that its creator is *not* a computer programmer. Nikki Yokokura, a lecturer in the Department of East Asian Studies, received a copy of Apple's HyperCard program in November 1987 but little dreamed where it would one day take her. Her curiosity about this programming tool was kindled when she heard of and read about some early HyperCard stacks. She began to test and explore some of these stacks and to investigate ways to create her own. Working entirely without a manual or documentation (Nikki assures us that she never uses documentation — unless she gets *really* stuck), she soon discovered how to access and manipulate the source code responsible for what appears on the screen. She looked at HyperCard stacks available on public-domain systems, tried and copied things that others had done, and finally learned to create simple stacks of her own.

Then, in January, Nikki conceived the idea of a stack designed to teach Japanese. It was only when she began to work on this, that she finally had to seek help from outside



lick appropriate buttons to
epeat or abort animation.

Figure 2: *Animated Visual Cues*

sources. She found Danny Goodman's book, *The Complete HyperCard Handbook,* very helpful for heavy-duty programming, and when she encountered problems, Nikki also received direction from programmers at UTCS. Lacking experience in mathematics or programming, she found some aspects of the work frustrating, but thanks to the relative ease-of-use of the HyperCard environment, by March she had come up with a visually exciting Japanese-instruction prototype.

This prototype was the starting point of what has since become KanjiCard, a full-blown software development project. Between January and June, Nikki put 600 to 800 hours into the project, taking the prototype through eight generations or revisions.

After more people were hired to work on KanjiCard, many new design changes were incorporated, leading to a design that is far more complex than the original. *Figure 1* shows the current KanjiCard directory, the starting-point of the application. Now, with 50 characters completed (each Kanji uses one megabyte of disk space), you can hear the recorded sound of each Kanji by itself, in phrases, and in sentences. You can view an animation of the brush-strokes used in writing each Kanji, or read a history and definition of each character and then see an animation that represents that history (*Figure 2* is a view of one of these "animated visual cues"). Six indexes (*Figure 3*) are available to help you find individual characters, and a "situation," consisting of a scanned photograph and a textual explanation, shows you how and where that character is used in daily life in Japan. Altogether, about 16 aspects of each character are covered.

To create some of the more captivating aspects of KanjiCard (for example, the sound recordings and the animations), Nikki had to use other applications outside of HyperCard. MacRecorder is a powerful sound-editing tool with which you can record and customize sound. The recording can then be put into a HyperCard stack sound resource, or

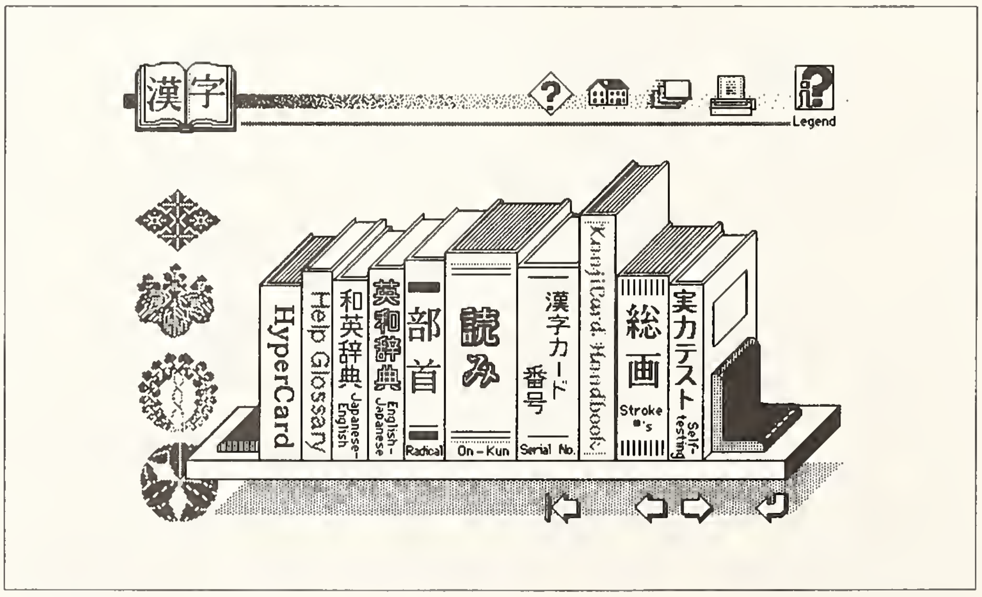


Figure 3: *Indexes*

incorporated into a VideoWorks animation sound file. VideoWorks is another separate application used to create animations for HyperCard stacks. Nikki found this a bit inconvenient to use because she had to exit HyperCard to alter work in VideoWorks, but she reports that a VideoWorks HyperCard driver is now available. It allows VideoWorks files to be played without exiting HyperCard background graphics, thus creating an illusion of 3D animation.

Now that KanjiCard is ready for field experiment, it will be shown at an upcoming computer fair in Toronto, possibly at a MacWorld Exposition in San Francisco in January, and possibly by Apple in the United States. For more information on KanjiCard, contact the Apple Research Partnership Program office at 978-7828.

BIB: A Bibliographic Application of HyperCard

Geoffrey Rockwell
rockwell@gpu.utcs.utoronto.ca

BIB is a collection of HyperCard stacks that keep track of bibliographic references, create formatted bibliographies, and provide note-taking facilities. Its emphasis is on the creation of formatted bibliographies for inclusion in other documents. For example, you might want to enter a collection of works used in a paper; BIB can prepare a consistently formatted, alphabetically ordered bibliography to be appended to the paper.

The BIB collection contains four stacks. The first is BIB, which keeps track of books and articles and produces the bibiliographies. You can create a bibliography in three ways: create it from selected cards in the stack; build it by adding entries one after another as you browse the stack; or create it from all the entries. The major limitation is the 32KB limit of HyperCard fields, since the formatted bibliography is put in a field for editing before it can be exported as a text file. This means that although you can reference an unlimited number of works, you cannot produce bibliographies longer than about ten pages. Once created, the bibliography can be edited and then either cut and pasted directly into a word processor, or printed from within HyperCard, or exported as a text file. BIB also allows you to both export to other BIB stacks and import selected cards from other BIB stacks.

To create the bibliography, BIB uses a predefined syntax to format each entry automatically. Book entries are all formatted in one way, as defined in the Syntax stack; articles in books are formatted in another predefined fashion. BIB does allow for unusual entries, however, by having an alternate feature where you can enter an alternative formatting. The alternate feature means that the number of entry forms can be kept to a minimum (Book, Article in Journal, Article in Book, and Other for unusual works).

BIB continued.

The second stack of the collection is Bib.Syntax. There you can change the syntax of the automatic formatting feature of BIB if, for example, you want to tailor the format of book entries to meet different publishing guidelines. Defining the formatting syntax is as easy as clicking buttons for the different fields and separators. At the moment, the stack has a major limitation — you cannot have different styles in the same bibliography, because HyperCard does not allow different styles in the same field. Thus you cannot ask BIB to italicize only titles; the whole entry must be in italics.

The third stack is Bib.Help, which once completed will provide two layers of Help: an initial *shortcuts* screen that appears when help is requested, and an online manual that explains the entire collection of stacks. As BIB is still under development, the Help stack is now only partially complete.

The last element is the Bib.Notes stack, which lets you take extensive notes on any work referenced in the BIB stack. Buttons link the two stacks. Once you have created the first note card for a work, you can jump back and forth between the reference in BIB and the notes in Bib.Notes. There are also ways to update connections severed when files are moved around and cards deleted.

The Bib.Notes stack is designed to take large numbers of notes on separate cards. A card is created for each logically separate note. Every card has a link to the original reference, along with a copy of the formatted bibliographic entry to which the note pertains. One feature lets you create pop-up notes on each card for subnotes in different fonts (such as quotations in Greek). Another feature can link a card to other cards in Bib.Notes besides those with the same reference. Thus, you could link a note on happiness in Plato's *Republic* to a note on happiness in Aristotle's *Metaphysics*.

You can browse the Note stack in a variety of ways, looking at the notes on a single work, at the first note for each work annotated, and at notes previously selected. You can even create bookmarks; when you click on one, it takes you back to the note marked.

The printing facilities of Bib.Notes allow you to print or export (for inclusion in other stacks) selected cards or selected cards in a particular order. You can print the cards as they appear on screen or print just the fields. You can also print the notes to disk for manipulation in a word processor.

The BIB collection does not compete with professional bibliographic packages, such as ***Pro-Cite*** , which have numerous entry forms and a programming environment for the formatting instructions. Instead it provides a simple but flexible way to keep track of references and to take notes. Because it is a HyperCard stack, it can be easily tailored to suit your needs. It is also free. The most recent stable copy is available from the Apple Research Partnership Program office (4 Bancroft Avenue, Room 213). Bring a disk and ask anyone in the office to copy it from our Mac II. Please pass on any suggestions as to how it can be improved or call if you need help adapting it to your needs. Our number is 978-7828. ▨

Early HyperCard Stacks

Mark Vanderkam
pubscoop@vm.utcs.utoronto.ca

We have a body on one of our computers — an electronic body for use as a teaching tool. Indicate any part on this body (using a pointing device), and information on and skeletal drawings of that area will appear. Created in HyperCard, **Electric Cadaver** is a *Navigational stack*, a presentation format designed to allow you to explore or navigate a stack of information. (For more on how HyperCard works and is used, see "HyperCard: Should You Be Interested?" on page 1)

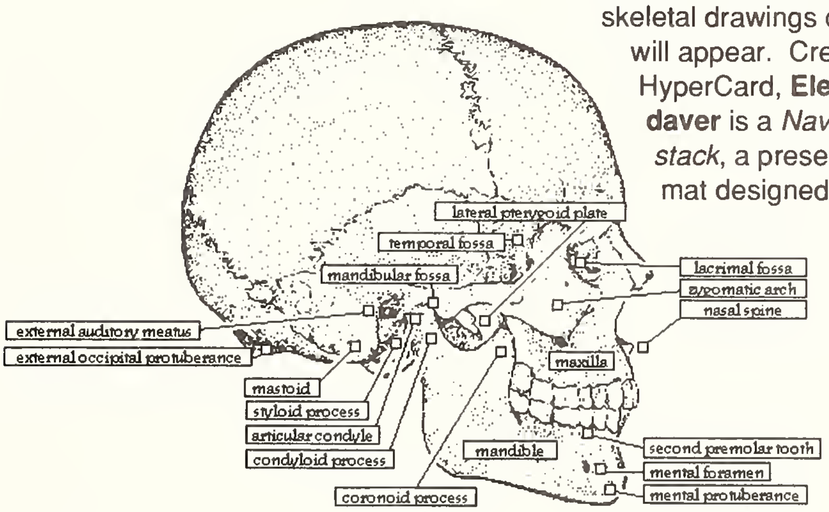


Figure 1: Skull with labels

This stack is a nice example of what can be done with HyperCard. When you first open the stack, a high-quality animation introduces the stack and its creators and shows a cadaver from a progression of internal and external perspectives. Finally a cadaver with a rather grim-looking skull appears, waiting for you to explore the stack. After this impressive opening sequence, we are left with a limited selection of drawings to choose from. Select the head and you can see an overall skull from the front, side, or top (the drawing can rotate). You can zoom in on the face or eye-socket. You can see a supposed x-ray view, and also a view with labels (*Figure 1*).

While the few drawings are relatively simple (the stack is a prototype and does not claim to be a definitive anatomy-resource), this stack could become a full-fledged information source on anatomy if the time and effort were invested in it.

Another stack, this one incorporating sound recordings with its graphics, is **Discover Canada**. This stack drew some interest at UTCS when it was first seen, because it was one of the first stacks available and it demonstrated some of the power and potential of HyperCard. Discover Canada presents a map of Canada. Select Toronto and a street map of part of downtown shows up. Select a hotel and a drawing of a bar appears while music comes over the computer's speaker. You can look at a few other cities, or even birds of Canada (four are depicted, along with good recordings of their songs).

In comparison to more recent stacks, Discover Canada seems a little disappointing because of the relatively small amount of information provided, but it is an excellent demonstration of the use of sound-recordings with HyperCard stacks. HyperCard is a relatively new computer tool, and Electric Cadaver and Discover Canada are examples of the kinds of information stacks under development since its release.

Subscription Renewals

It is now time to renew your subscription to ComputerNews. Because all subscriptions run from January to January, even new subscribers must renew at this time, if they wish to continue receiving ComputerNews.

*To renew, just fill out the **Annual ComputerNews Subscription Renewal Form** appearing opposite this article. Any subscription not renewed by the beginning of January will be dropped.*

Please remember that this policy helps us keep our mailing costs down and continue to send you the newsletter free of charge.

Return of Domesday and Much More

Mark Vanderkam
pubscoop@vm.utcs.utoronto.ca

Domesday and two new eye-catching videodisc tours will be displayed at UTCS for two weeks in January. During that time, we will also feature two demonstrations of this interactive videodisc system and a talk on an exciting computerized information project under development in Canada.

Displayed at UTCS last winter, Domesday (pronounced *doomsday*) is a computerized multimedia tour of the United Kingdom. Dozens of people came in to see this presentation, and many were impressed that so much information is available in such an easy-to-use and visually captivating format. It is no wonder it's impressive: the BBC spent £3,000,000 (about $7,000,000) to create Domesday.

To say this system is easy to use may be an understatement. Every schoolboard in the U.K. bought copies because it is such a good learning tool for children. Sitting before the screen, with clicks of a button you can call up full-colour slides as clear as printed photographs. There is an hour of British news video-footage, 55,500 still photographs (including satellite photos), 24,000 maps (including street maps of many areas), and 150,000 pages of text and 9700 sets of data on life in the U.K.

This information, the equivalent of two sets of the *Encyclopaedia Britannica*, is stored on two LP-sized interactive videodiscs. The combination of information formats gives users a very comprehensive and detailed survey of the U.K.

Two new discs

The ecology and volcanoes are the subjects of the two new discs, the only ones released since the Domesday discs. **Ecodisc** allows senior high school and first-year university students to create five-year plans for managing a wetland habitat. Students can look at maps, video footage, and still photos (winter or summer shots are provided, as well as underwater and microscopic views); put radio collars on animals to trace their movements; consider letters and information sent in by wildlife groups, industrialists, and concerned citizens; seek advice from experts, and then manage the habitat.

Volcanoes, created through a joint venture by the BBC and Oxford

University Press and touted as an "exciting, explosive interactive videodisc," brings you a wealth of information on volcanoes and volcanism. Like the other videodiscs, Volcanoes uses sharp photos, dramatic film footage, animated maps and diagrams, and text files to present a comprehensive tour of the subject. This videodisc also uses hypertext, which links definitions with highlighted words. Point the screen pointer at a highlighted word, click a button, and up comes the definition for that word. Through the "Find" facility, complete definitions from the *Oxford English Dictionary* for 1500 words associated with volcanism are available. You can also investigate six major volcano case studies, including Mount St. Helens, Nevado Del Ruiz (which erupted in Columbia in 1985 and killed 25,000 people), and Io, Jupiter's volcanic moon.

This BBC interactive video system continues to excite new interest and support. Since UTCS last had the system, Olivetti has displayed it at major mapping conferences, at the University of Waterloo, and at York University. The Markle Foundation in New York reportedly gave US$1,000,000 to support the work, and, as we reported in March, other countries are interested in producing similar videodisc tours.

Beyond Domesday: the next generation

Statistics Canada and the Canadian Studies Programme of the Secretary of State is investigating the possibility of a Canadian adaptation of this interactive video system. They feel Domesday is an important groundbreaking step in presenting the power of a multimedia database to a general audience. Building on what the BBC has done, they are currently planning a national survey called the **Jean Talon Project** (named after Canada's first census taker) and **Who We Are**, an Ontario module for the national project.

Booking Information

Demonstrations of Domesday, Ecodisc, & Volcanoes:

January 12, 4:30 - 5:30 p.m.
January 16, 2:00 - 3:00 p.m.

To book a spot, call 978-4565.

Booking Personal-Time with the System.

Available from January 10 to January 24, between 10:00 a.m. and 6:00 p.m.

To book a spot, call 978-4565.

Jean Talon & Who We Are: Mapping the Future

Mr. Glickman will speak on January 17, 2:00 - 3:00 p.m.

To book time, call 978-5601.

A feasibility study for the Jean Talon Project was done with funding from a collaboration of federal and provincial organisations including the National Film Board (which is working on "A Video Encyclopedia of Canada in the Twentieth Century," a project that will include 15 hours of archival film which could be committed to videodisc for inclusion in Jean Talon), the Canadian Museum of Civilization, and the Ontario Ministry of Education. The Jean Talon Project, it is hoped, will encourage the creation of videodiscs by private and public interests and will contribute to the development of videodisc technology in Canada. To facilitate widespread interest, Jean Talon is being designed as an open-architecture system to allow people representing a broad range of interests to eventually develop their own modules.

During the exhibition at UTCS, the Director of Statistics Canada (Ontario Region), Mr. Victor Glickman, will give a talk on the Jean Talon Project and the Who We Are module. He will present an overview of where these projects stand now and of where they are going in the future. See the box for booking information.

Demonstrations Scheduled

We have scheduled free demonstrations of the BBC system on two dates.

You can also book time to come in and use Domesday, Ecodisc, and Volcanoes personally. The UTCS advisors will be available to help you with the system.

Macintosh Virus Prevention

Dariusz Dabek
dabek@vm.utcs.utoronto.ca

Computer viruses have attracted a lot of media attention in recent months. Although we have not seen any symptoms of a viral epidemic at U of T, it is still important to be protected. This article describes and explains some of the tools and procedures that can be used to protect your Macintosh computer. Currently, UTCS has three such tools available: **Virus RX**, **Interferon**, and **Vaccine**. They can be obtained directly from UTCS Advising Services during business hours or from the UTCS Bulletin Board System (BBS).

A virus is simply an unauthorized program that has been injected into a user's computer. Such a program usually replicates itself every time the computer is used. In some extreme cases, the virus attempts to damage the user's data.

Most often, viruses are contracted when you retrieve infected programs from a bulletin board over a modem or when you use an infected floppy disk. The first rule in virus protection is to check any new programs obtained for infections. This rule is strictly followed when we add new programs to the UTCS BBS; each new piece of software is examined on a separate test computer. Home or office users can do this on a machine without a hard disk or with the hard disk switched off. Look for any suspicious files created after running the program for the first time. Check the size of the System Folder before and after the first run.

The above procedure detects almost all common virus programs; it should be performed on all new disks, including commercial programs.

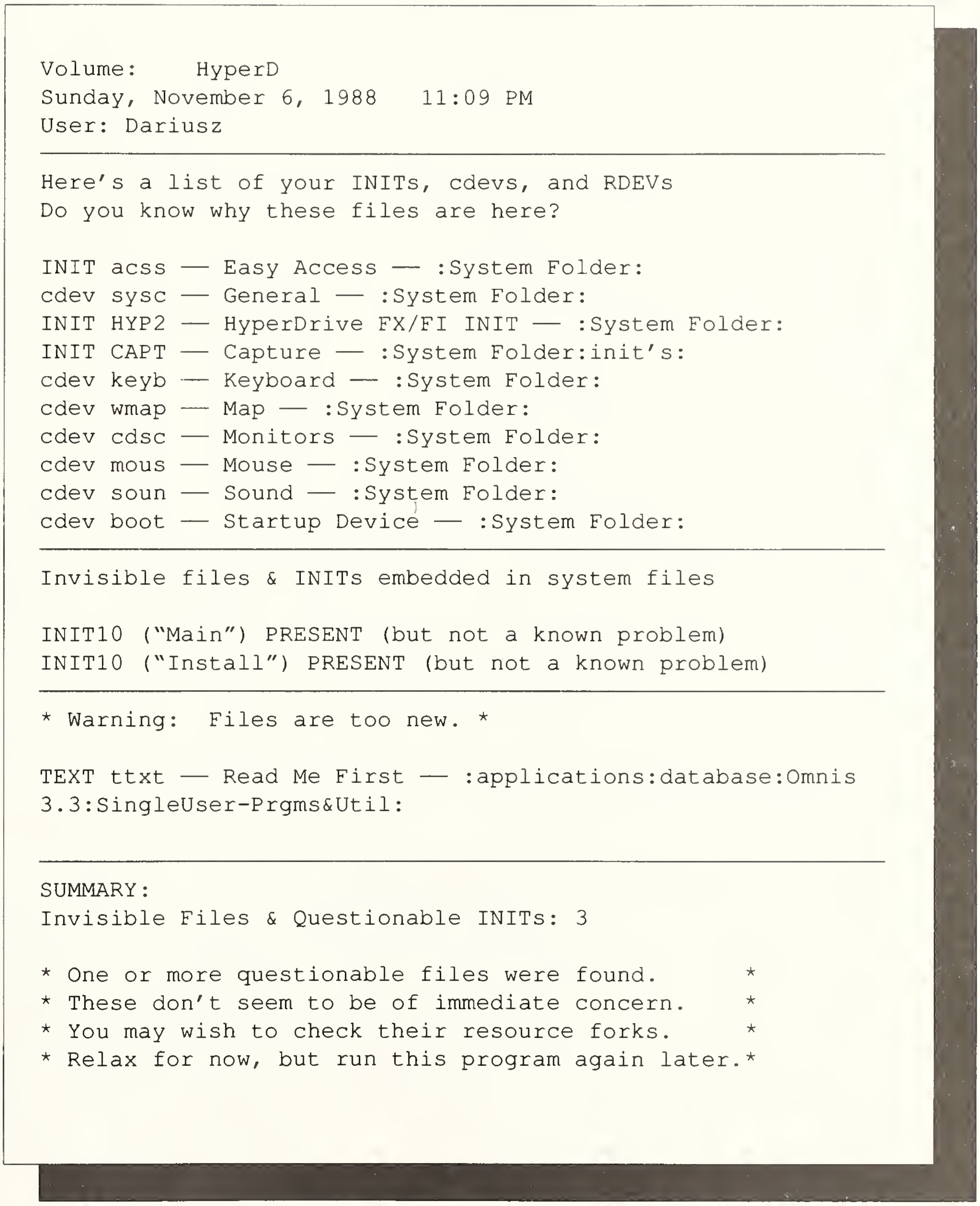
To protect the machine further, you can install programs that detect known viruses or attempt to prevent them from multiplying. Virus RX and Interferon are designed to search for or remove viruses from infected files. Interferon can do both of these actions in a simple procedure; after you double-click the program icon and select the hard disk for examination, a report is produced indicating where the viruses, if any, are present. This type of program should be run frequently. For a 60 megabyte drive, it takes less than a minute.

Reports provided by these programs may seem rather confusing. *Figure 1* is an example of a report generated by Virus RX. Casual users should be concerned only with any unexpected differences between the current report and the last one.

Vaccine is another type of program, an INIT; it resides in the System Folder and is loaded every time the Macintosh is turned on. Its purpose is to keep the System Folder, especially the System file, from being modified. This is the most common file a virus attacks when attempting to multiply. Vaccine warns you whenever any program tries to modify the System file. The disadvantage of this program is the frustrating slow-down of the computer's operation due to the continuous monitoring of the System Folder.

Viruses are dangerous, but they can be stopped by using the simple procedures described above. Taking less than a minute to check a hard disk for infections can save you hours of work. These preventive programs are in the public domain and may be distributed freely to friends or to people in your office.

For further information, please contact Advising at 978-HELP. ▨



```
Volume:     HyperD
Sunday, November 6, 1988   11:09 PM
User: Dariusz
____________________________________________________________
Here’s a list of your INITs, cdevs, and RDEVs
Do you know why these files are here?

INIT acss — Easy Access — :System Folder:
cdev sysc — General — :System Folder:
INIT HYP2 — HyperDrive FX/FI INIT — :System Folder:
INIT CAPT — Capture — :System Folder:init’s:
cdev keyb — Keyboard — :System Folder:
cdev wmap — Map — :System Folder:
cdev cdsc — Monitors — :System Folder:
cdev mous — Mouse — :System Folder:
cdev soun — Sound — :System Folder:
cdev boot — Startup Device — :System Folder:
____________________________________________________________
Invisible files & INITs embedded in system files

INIT10 (“Main”) PRESENT (but not a known problem)
INIT10 (“Install”) PRESENT (but not a known problem)
____________________________________________________________
* Warning:  Files are too new. *

TEXT ttxt — Read Me First — :applications:database:Omnis
3.3:SingleUser-Prgms&Util:

____________________________________________________________
SUMMARY:
Invisible Files & Questionable INITs: 3

* One or more questionable files were found.      *
* These don’t seem to be of immediate concern.    *
* You may wish to check their resource forks.     *
* Relax for now, but run this program again later.*
```

Figure 1: Sample report generated by Virus RX

ARCHIVIST: Database Manager and Text Processor

Paul Roth
proth@vm.utcs.utoronto.ca

ARCHIVIST, from IRL Press Limited at Oxford, is a PC-DOS/MS-DOS textual database management system. It lets you organize lecture notes, research notes, product information, bibliographies, or any other text-based information that can be arranged in discrete units. It should appeal to academics who need to search a set of abstracts or notes for specific occurrences of certain words or phrases, with minimal regard to how the textual data is organized. As a bonus, bibliographies can be constructed from the text database, with relative ease and in your choice of a number of possible standardized formats.

ARCHIVIST comes on two 360 kilobyte diskettes. It includes the main database program, an integrated text editor that uses WordStar-like commands to edit ASCII text files, a few small utility programs, and a cast of supporting ASCII text files that you may change using any ASCII editor or word-processing package able to handle them. A small-sized looseleaf user manual accompanies the two diskettes. We reviewed Version 1.00.

When you first install ARCHIVIST, by using the DOS **Copy** command, you have a sample database at your disposal, which enables you to familiarize yourself with most of the program's features. You enter the package by typing **AR**, which displays the main ARCHIVIST screen. The "look and feel" is akin to Microsoft Works, or even MS Windows (with no mouse): a menu bar is at the top of the screen, a status line at the bottom. Choosing a menu option (by pressing <Alt><x>, where "x" identifies a particular option) brings you to a drop-down menu, from which you use the cursor arrow keys, either vertically to choose an option or horizontally to choose other menu-bar options offering their own drop-down menus. It takes only a short time to become familiar with this syntax.

Online Help is available by pressing <F1>. This is context-sensitive, so that you receive Help about the current operation. Alternatively, you may ask for general Help, which is organized by topic. The Help windows that pop up contain useful information, explained in clear English, and in most cases add more insight than the user manual alone.

Unlike dBASE, records are stored "somewhere" in a database. There is no concept of record numbers; you are "merely expected to be aware that there are a lot of records stored somewhere." The idea is apparently to make life easier for the user: just enter your records without worrying about where they go, knowing that you can find them easily enough when the time comes.

Another strength of the package is its ability to mix record types in a single database. Thus you might have, in the same collection, miscellaneous notes along with references to a number of journal articles and books. These can all coexist in the same database. Subject to disk-storage and memory limitations, each record may contain up to 16300 free-form characters (roughly eight typewritten pages). The record may be divided into virtually any number of fields. These fields are limited in size only by the size limit of the record and typically include such items as author, title, date, and abstract. Each record type could consist of one or all of these fields, and each could contain its own other fields as well.

Searching the database can be done by looking for occurrences of a given string anywhere in the database, or anywhere in the database in a certain field. This is a useful feature, to be sure.

Searches are done quickly because ARCHIVIST maintains its own index of the database. Every word is indexed, with the exception of stop words defined in STOPWORD.LST, an ASCII file that you can modify. A nice feature is the ability to identify synonyms, by modifying another ASCII file called SYNONYM.LST. For instance, it can contain lines like:

```
baseball=pitcher,catcher,inning,
 abstract=play,Yankees
```

Thus, a search for “baseball” includes a search for “baseball”, “pitcher”, “catcher”,”inning”, or “Yankees” anywhere in the database, or the term “play” if it exists in the field called “Abstract”.

More than one search term can of course be specified: this is done by “Narrowing” or “Widening” the search, roughly equivalent to the Boolean AND and OR respectively. The result of a search is a *found list*, which can be perused through the Function keys. To zero in on a subset of this found list, you narrow the search; to expand it you widen it. You can exclude (Boolean NOT) the terms as well: this is called Selective Drop. In almost all cases, searching is easy to do because of the way in which function keys and/or drop-down menus are used.

ARCHIVIST allows you to create macros, which are recorded search requests that you can “replay” at another time. Thus if it takes 15 “Narrows” to isolate a found list, and you have recorded that series of actions, a simple invocation of the macro via a function key is all you need to do the same search in the future. Macros are stored in an editable ASCII file.

If you have ever used WordStar, you will be at home with the editing capabilities of AR and its editor. If not, they are easy enough to learn. Editing the supporting ASCII files can of course be done in other word-processing programs, or even with EDLIN. In practice, you would probably prefer a package that edits straight ASCII directly (you don’t want to use wordwrap in the synonym list, for example).

One of the files you must create yourself is for Imports. This is used to bring in data from outside the package. You can of course enter your own records, but most people also want to be able to load a number of records from an external source such as dBASE or an online information system like MEDLINE. This part of ARCHIVIST can be a little tricky for the nonprogrammer, as it requires some cryptic definitions to get things just right. Similarly, you must create an Export format file if you wish to export files. This too can be tricky for the nonprogrammer. ARCHIVIST provides sample import and export files together with detailed explanations of printer control codes and the UNIX-like symbols and wildcards used to describe field and record locations. However, some trial and error is still needed to achieve total success with exports. (There could be a way around this problem; see below.)

Exports can go to the printer or to a file. The file includes any printer control codes you inserted, and again is just a modifiable ASCII file. The weakest part of ARCHIVIST is in its printer support; it deals basically with Epson printer control codes, but you can also edit ARCHIVIST’s configuration file (yes, it too is in ASCII) to define printer control codes for your particular printer. No PostScript support is available. The developers apparently expect that if, for example, you want laser output, you will create an ASCII file to be loaded into your normal word-processing program. This is unfortunate, but still workable.

As mentioned above, there could be help for nonprogrammers worried about creating import and export format files. If enough interest exists for ARCHIVIST, UTCS could obtain a site license or volume discount arrangement from IRL Press. In that case, there would be a number of people on campus using the package, and we expect that they would share their import and export format files with others (through UTCS and/or the UTCS Bulletin Board Service). Thus the trailblazer who figures out the precise definition for CompuServe journal references, for instance, could pave the way for others to use that data too. Similarly, we would distribute standardized export files for all developed bibliographic methodologies (ARCHIVIST comes with Harvard and Vancouver defined).

A demonstration version of the program is available in the IBMNEW or IBMWP areas of the UTCS Bulletin Board Service; it is called ARCHDEMO.ARC. This demo is a full version of the program, but is limited to 16 records per database.

Please try it out and let us know if you would like to acquire this product. If enough people seem interested, we will pursue the site license arrangement. ARCHIVIST currently sells for US$95; under a license this would probably be reduced to about CAN$30 per copy. The entire user manual comes on a diskette, with an optional printed manual sold at extra cost.

We have been discussing Version 1.00. Version 1.02 of ARCHIVIST, now shipping, has a number of improvements. Documentation includes an electronic manual that is itself a searchable archivist file. Phonetic searches are also possible; e.g., a search for "Smith" also finds "Smythe", "smyth", etc.

ARCHIVIST seems like an excellent product that could be useful for both the casual notetaker and the serious bibliographer. The program is reasonably well documented, has good Online Help, and provides genuine functions at a good price. It appears to be robust; in all our tests, it never crashed or hung the computer. Try it today, and if you're interested, please call Paul Roth at 978-5426. ▒

COMMODORE PC 60-40

Paul Roth
proth@vm.utcs.utoronto.ca

The Commodore PC 60-40 represents the high end of the Commodore line of computers: an 80386 machine, German-made, providing a solid platform for the future. While not running at blinding speed, the PC 60-40 provides enough punch to satisfy most people's needs.

The machine we reviewed contained a 44 megabyte (MB) Rodime hard disk, which we benchmarked as having an average access time of 26 milliseconds, with a data transfer rate of about 250 kilobytes (KB) per second. The disk is actually a 3.5-inch device mounted on a 5.25-inch chassis, taking up one of the four available half-height 5.25-inch drive bays. Thus the 60-40 can optionally contain two hard disks, a 1.2MB floppy disk drive, and a 1.44MB 3.5-inch floppy disk drive. (A PC 60-80, with an 80MB hard disk, is also produced by Commodore but is not available in Canada.)

The hard disk controller is not built in; rather it sits in a 16-bit slot and can handle four devices, as already mentioned (two fixed disks, two floppy). All drive cables (and power cables) you might need for expansion are provided, a useful feature. The controller is made by Western Digital and looks very clean, except for some rough card edges on the circuit board itself.

Memory is treated in a somewhat unusual way. You are given 2.5MB standard, consisting of 512KB RAM on the motherboard, with a 2MB RAM board occupying a 32-bit slot. All the RAM comes as 256KB, 120 nanosecond chips, with parity. The memory board, made by Intel, looks very clean. While the more modern technology for memory is SIMMs (Single In-line Memory Modules), Commodore apparently has a stock of 256KB chips, which it is using up during a current RAM shortage in the market. As a result, you pay less but get slightly less modern technology.

The most unusual feature of memory usage, however, is the way in which RAM is apportioned for DOS. By default, you have 512KB RAM, with 2MB of extended memory that you could use, for example, as a 2MB RAM disk. To get the full 640KB for DOS, however, you use a device driver called CTRL386.SYS; it is configured by using the CONTROL/386 software, which is written by Phoenix Technologics and included with the PC 60-40. This software lets you arrange extended memory as you wish and contains a number of performance enhancements. You can set up part of RAM for a disk cache; copy the system and EGA BIOS from ROM to RAM; and allow for "automatic diskette slowdown," which temporarily slows the processor speed when the floppy disk is accessed directly. The last is useful for timing-dependent programs such as copy-protection.

If you wish, CONTROL/386 lets you define all of the extended memory as expanded memory, conforming to the Lotus/Intel/Microsoft (LIM) specification. You can also configure the device driver so that it is loaded automatically when you boot. Alternatively, you can choose to be prompted at bootup as to whether or not to use the driver. This could be useful in situations where you are alternating between configurations. The documentation for CONTROL/386 is a README text file on diskette. It is brief but clear enough for the experienced; novices may require some help.

Manufacturer: Commodore Business Machines Limited

Type: 32-bit MS-DOS.

System Unit: 80386 CPU, with slot for 80287 or 80387 numeric coprocessor; 2.5MB RAM standard; two 9-pin serial ports; 2 parallel ports; 8 expansion slots; 16MHz clock speed with 3 wait states; On/Off switch at rear; clock/calendar; keylock in front of unit; 220W CSA power supply with 110/220V autoswitching; monitor socket at rear of unit.

Keyboard: 101-key IBM AT-style layout, with large indicator lights for "CapsLock" and "NumLock"; fairly small "Enter" key; separate "Enter" key on numeric keypad; lightweight design; cord attaches at rear of system unit.

Monitor: Commodore 1084 RGB Colour monitor was reviewed; contrast, colour, and brightness controls within easy reach of operator on front of monitor. (A more usual configuration would be an EGA monitor.)

Disk Drives: One 5.25-inch1.2MB floppy drive, plus one 44MB internal hard disk.

Physical Description: 6.25"H, 21.0"W, 16.5"L; approximately 30 lbs; access to inside of machine: 3 screws on the back panel and 2 screws on each side of the system unit; metal casing.

Warranty: One year warranty.

Compatibility Level: 9x%.

Bundled Software: MS-DOS Version 3.21; GW-BASIC 3.20.

Documentation Provided: User Manual (approximately 45 pages), *MS-DOS 3.2 User's Guide and User's Reference* (350 pp), and *GW-BASIC Interpreter User's Guide and User's Reference* (300 pp).

The **keyboard** is lightweight, with a slightly rubbery feel, and some auditory feedback. The only problem is that the feet slide on the desktop but Commodore claims to be correcting this.

The inside of the system unit presents no problems. The cover slides off and on easily enough as long as you take normal care to avoid jostling the cables. The motherboard is made by Intel. The power supply is large and can automatically switch between 110V and 220V circuits, so using the machine abroad should present no problems. There is a powerful fan with the power supply, necessary as heat build-up must be constantly prevented. The fan contributes a normal amount of background noise during use.

A readily accessible battery is mounted with Velcro on the inside of the rear cover of the sytem unit. This can be easily replaced when necessary, perhaps every four years or so. The system unit can be locked (as per normal AT machines). A speaker is housed in the front of the unit.

The eight **expansion slots** consist of one 32-bit proprietary slot, occupied by the 2MB RAM board as mentioned above; one free 32-bit slot; one 8-bit slot taken up by a parallel/serial card; one free 8-bit slot; two free 16-bit slots; one 16-bit slot occupied by the disk controller; and one 16-bit slot taken up by the graphics adapter, an ATI EGA Wonder card (which can also be placed in the 8-bit slot).

The **EGA Wonder** is a flexible, Canadian-made adapter offering MDA, Hercules, CGA, CGA emulation, Plantronics, and EGA, on a composite, TTL monochrome, RGB colour, or EGA monitor. Software controls which of the modes the adapter uses at any given time, although DIP switches on the card itself determine which mode the card uses at power-on time. The DIP switches are easily accessible at the rear of the system unit, without removing the cover.

The EGA Wonder card has a good enough reputation and appears to work well. It does have a problem when you use EGA mode with an RGB monitor, as we did for this review. Because the EGA Wonder uses interlacing, a constant flickering is produced on the screen in some packages, such as Windows. This can be alleviated by turning down the contrast, but even then the image quality is marginal. If you plan to use software that operates in graphics mode, as Windows does, an EGA monitor would be a much better choice, even at the additional cost. With an EGA monitor, the flickering problem is eliminated completely and the screen display is quite attractive.

We tested the PC 60-40 with a number of **hardware devices** and found not the slightest problem! The machine appears to have excellent board compatibility. Devices tested include a Bernoulli Box, CD-ROM player, Logitech serial mouse, and network adapters.

The **Bernoulli Box** consists of two 20MB removable cartridge drives housed in an external chassis, connected to a board that plugs into a free expansion slot. We simply took the board from an AT, loaded the Bernoulli software for the appropriate device driver, and tried using the cartridges. They worked fine first time, showing none of the problems with DMA channels or interrupts that are sometimes encountered with less compatible machines. Among other software, we ran SAS/PC from the Bernoulli Box; it worked perfectly.

We attached an **Hitachi CD-ROM player** to the PC 60-40. It too worked fine first time. We tested it with the PC-SIG CD-ROM, and with Microsoft Bookshelf (under MS-DOS CD-ROM extensions). Both products performed flawlessly.

Network adapters also worked well. We had no problems connecting to a **Token Ring Network** under **Novell Netware**, simply taking the IBM Token Ring Network adapter out of an existing network workstation and popping it into the PC 60-40 without even resetting any switches. On the network, all application software we tried worked fine, including Lotus 1-2-3 and the network versions of WordPerfect and dBASE III Plus.

Similarly, we used a Centram **TOPS** board to connect over Appletalk to Macintoshes running TOPS network software. Again, no problems were encountered, as files transferred easily to and from the Macintosh. As TOPS often requires some fiddling to get its switch settings correct, this was a useful test to pass first time.

Software compatibility appeared fine. The usual high-profile products such as WordPerfect 4.2 and 5.0, dBASE III Plus, and Lotus 1-2-3 worked well, as did a number of public-domain and shareware utilities and PC-DOS 3.2. Lotus Agenda, Manuscript, and Metro all ran without apparent error, as did ProComm.

The **GWBASIC** used allows various screen modes, and supports EGA monitors. It is well described in the documentation, which is similar to that of the PC 20-III (reviewed in the July/August *ComputerNews*). It is suitable for both the novice and the experienced user.

Summary

The Commodore PC 60-40 appears to be a worthwhile entrant in the moderately-priced 80386 market. The 16MHz speed, use of 256KB chips, and EGA display adapter are not state-of-the-art but provide value for the money nonetheless. The performance enhancements in the CONTROL/386 software, together with the excellent hardware and software compatibility, make this machine worthy of consideration. ▨

Microcomputer Courses for the New Year

– Irene Rosiecki

The course schedule is now ready for anyone interested in attending one of our noncredit microcomputer courses in the new year. Although the following list includes only the courses scheduled until the end of March, courses are scheduled until the end of June. For information on the schedule after March, or to register for a course listed below, please call 978-4565.

Microcomputer Concepts (PC)
January 11, 1989
1:00-4:00 p.m.

Introduction to dBASE III Plus
Jan. 30-Feb. 3, 1989
2:00-5:00 p.m.

Desktop Publishing Seminar
February 16, 1989
10:00-3:00 p.m.

Introduction to DOS
Jan. 30-Feb. 3, 1989
12:00-2:00 p.m.

Electronic Mail Seminar
January 18, 1989
11:00-12:30 noon

Introduction to Excel on the Macintosh
January 23-27, 1989
2:00-5:00 p.m.

March 6-10, 1989
9:00-12:00 noon

HyperCard Introductory Seminar
January 18-19, 1989
10:00-12:00 noon

Introduction to Lotus 1-2-3
February 13-17, 1989
2:00-5:00 p.m.

March 27-31, 1989
9:00-12:00 noon

Introduction to PageMaker
February 20-24, 1989
9:00-11:00 a.m.

ProComm Communications Seminar
January 18, 1989
9:00-11:00 a.m.

Introduction to SAS on the PC
January 23-27, 1989
3:00-5:00 p.m.
March 6-10, 1989
3:00-5:00 p.m.

Intermediate Applications of SAS on the PC
February 20-24, 1989
2:00-5:00 p.m.

Introduction to Microsoft Word (IBM PC)
Jan. 30-Feb. 3, 1989
9:00-12:00 noon

Introduction to Microsoft Word (Macintosh)
January 23-27, 1989
9:00-1:00 (Jan.23)
9:00-12:00 (Jan.24-27)

March 27-31, 1989
9:00-1:00 (Mar.27)
9:00-12:00 (Mar.28-31)

Introduction to WordPerfect Version 5.0
January 23-27, 1989
12:00-3:00 p.m.

February 13-17, 1989
9:00-12:00 noon

March 6-10, 1989
9:00-12:00 noon

March 13-17, 1989
12:00-3:00 p.m.

March 27-31, 1989
12:00-3:00 p.m.

Advanced WordPerfect Version 5.0
February 13-17, 1989
12:00-2:00 p.m.

Introduction to WordPerfect Version 4.2
January 23-27, 1989
9:00-12:00 noon

March 13-17, 1989
9:00-12:00 noon

Don't delay — most courses fill up quickly.

Site License for SYSTAT on Macs

Joyce Bernstein
jbernst@vm.utcs.utoronto.ca

University of Toronto Computing Services is completing arrangements for a University-wide site license for SYSTAT on Macintosh computers. Our interest is partially in response to users' requests for such a site license. There is sufficient interest for us to proceed with an initial site license, but more participants are needed to make cost sharing viable. The initial fee for SYSTAT is $115, plus $40 for manuals. The license will be renewed each July 1.

SYSTAT is currently the only full-feature statistics package available for the Macintosh. Although SYSTAT was originally developed for DOS, the move to the Macintosh world has been reasonably successful. The current implementation takes advantage of the standard desktop environment. However, the actual statistical computations are still coded in a linemode style reminiscent of DOS. The same manual is used in both environments.

This situation will change for the better in the summer of 1989. SYSTAT has announced that Version 4.0 for the Macintosh will then be available. Version 4.0 will have a separate Mac-specific manual, and the user interface is said to take full advantage of the Mac environment. For participants in the site license, an upgrade to Version 4.0 is included in the initial fee.

The following provides an overview of SYSTAT capabilities. The general linear model module of SYSTAT can do simple, multiple, and stepwise regression; ANOVA, ANCOVA, MANOVA (including unbalanced, missing cell, nested, and repeated-measures designs); and discriminant analysis. It also provides diagnostics for influential cases, autocorrelated error, and collinearity. Other SYSTAT modules produce descriptive statistics, t-tests, correlation matrices, nonparametric statistics, cross-tabulations, log-linear models, time-series ARIMA models, principal components analysis, multidimensional scaling, factor analysis, cluster analysis, and nonlinear estimation. Statistical graphics include scatterplots, histograms, box plots, stem-and-leaf plots, probability plots, and series plots.

The SNAC group in the UTCS Information Centre has a copy of SYSTAT 3.2 on a Macintosh SE. Our experience with this version has been very positive, and Version 4.0 can only make things better. To arrange to preview SYSTAT or to acquire a copy of SYSTAT 3.2 through the site license, call Joyce Bernstein at 978-5128.

There are no current plans for a site license for SYSTAT for DOS machines, but we will consider getting a site license if there is sufficient interest. Please give us a call if you are considering acquiring SYSTAT to run under DOS. ▒

SAS Surcharge for External Customers

Ron Vander Kraats
rvk@vm.utcs.utoronto.ca

Effective October 26 (the beginning of the November accounting period), the use of SAS on our VM/CMS service by customers *not* affiliated with the University is subject to a 25% surcharge. The license agreement between the SAS Institute and the University of Toronto stipulates that the uses of the SAS software package by non-University customers be subject to a royalty of 25% of CPU charges incurred during the running of SAS programs.

A separate accounting record for each time SAS is used will appear on the CHARGES accounting report for all customers.

Numerical and Scientific Software for Microcomputers

Andrzej J. Pindor
apindor@vm.utcs.utoronto.ca

PC Scientific FORTRAN Mathematical Subroutine Library

With the increasing power of microcomputers, more and more people are moving their computations from mainframes to micros. Naturally, they would like to tap into a large body of sophisticated software already used on mainframes. PC Scientific Inc. (St. Paul, Minnesota) has adapted a number of mainframe scientific subroutine libraries to a microcomputer environment and offers them on a commercial basis:

MINPACK1-LIB, derived from Argonne National Laboratory's MINPACK-1, solves nonlinear systems of equations and nonlinear least-squares problems.

FITLIB, derived from Dr. Alan Cline's FITPACK, fits curves or surfaces through a given set of data points, using tension splines.

SPARSGEM, derived from SMPAK (Scientific Computing Associates), solves sparse systems of linear equations.

QUADLIB, derived from QUADPACK, performs numerical integrals.

ODELIB, derived from Lawrence Livermore Laboratory ODEPACK, solves systems of first order differential equations (has automatic stiffness detection).

Each of the products offered by PC Scientific Inc. comes with detailed manuals and interactive tutorials, and with "skeleton files," which make it easy to write programs using the provided subroutines, even for people with limited experience in FORTRAN programming. The libraries are provided for all major FORTRAN compilers.

We have received from PC Scientific Inc. demonstration diskettes for all of their products except ODELIB. If you are interested, please come and try demonstration diskettes for yourself.

In general, these products may appeal to people with limited programming experience and who face very specific numerical problems that are covered by one of the libraries. The interactive tutorials and skeleton files, as presented on the demo diskettes, might save you some of the time and effort needed to familiarize yourself with the software, compensating for the price (see box).

Vendor: PC Scientific Inc., 6 Pine Tree Drive, Suite 250, St. Paul, Minnesota 55112. Phone: (612) 490-0615.

System Requirements: IBM PC, -XT, -AT, portable, or PS/2 microcomputer; one of the following compilers: IBM Professional FORTRAN 1.30, Lahey FORTRAN 2.22, Microsoft FORTRAN 3.31 or 4.01, Ryan-McFarland FORTRAN 2.42; the version of DOS required by the FORTRAN compiler used; hard drive recommended but not required; at least 256KB of RAM, 640KB recommended; a math coprocessor chip not required but strongly recommended.

Price: Each library costs US$350.

continued...

muMath-83 Symbolic Math System

muMath-83 is a computer algebra package and as such is very different from other numerical software packages already reviewed in *ComputerNews*. Computer algebra packages in general are discussed in the next article, and those with no exposure to these programs are strongly advised to read it in conjunction with the present article.

muMath-83 is the smallest of the available symbolic-manipulation packages. It has been designed with small microcomputers in mind and can run on many different microcomputers. For IBM compatible micros, you can manage with as little as 128KB RAM.

Capabilities of muMath-83 include:

- exact rational arithmetic
- algebraic and trigonometric manipulations
- symbolic equation solver
- matrix operations
- symbolic differentation and Taylor series
- symbolic integration
- closed-form summation and products
- ordinary differential equations

The system is organized into 22 small modules; you load only those needed for the particular capabilities you want. If you have more than 320KB RAM, you can load the whole system.

It might be expected that within each of the above mentioned fields, abilities of muMath would be inferior to those of larger computer algebra systems, but this does not always hold true. For instance, we have found that it can do some integrals that larger packages, such as REDUCE and Maple, cannot perform.

The most visible limitation of muMath-83 is its inability to perform full factorization of algebraic expressions. Although understandable for such a small system (full factorization is algorithmically a nontrivial problem), it considerably affects the usefulness of the package. Another important deficiency is a poor DOS interface.

The system includes two sets of interactive online lessons. The first of these sets explains features of muMath, while the second teaches the muSIMP programming language, used for writing muMath modules. muSIMP can be used together with the resident display-oriented editor to extend muMath or to create different applications.

The manual is well organized and covers in detail both muMath usage and muSIMP language.

In conclusion, muMath is a small symbolic-manipulation program that can be run on machines with very little RAM. Nevertheless, it has a reasonable range of capabilities and can be helpful in solving a number of routine mathematical problems. It might also serve as an educational tool for familiarizing yourself with problems related to implementing symbolic mathematics on computers.

Vendor: Soft Warehouse Inc., 3615 Harding Avenue., Suite 505, Honolulu, Hawaii. The package can be obtained from a number of software distributors, such as Programmer's Connection.

System Requirements: an 8086 or 8088 microcomputer running MS-DOS or PC-DOS operating system; minimum 128KB of RAM (it can take advantage of up to 320KB); at least one standard floppy drive; muMath can run also on Apple (II, IIe, and III) and CP/M-80 microcomputers.

Price: IBM PC version costs US$229 (at Programmer's Connection)

Assorted Public-Domain Software

We have received several integration routines in FORTRAN for one- and two-dimensional integrals of real and complex functions (simultaneous integration of real and imaginary parts gives much better results than two separate integrals). These are fast and reliable adaptive routines that can handle difficult functions. Routines are written in standard FORTRAN and should compile with any compiler (in fact we compiled them with Microsoft FORTRAN 3.3, IBM FORTRAN/2 and Lahey FORTRAN 2.22). For more information, please call Andrzej Pindor at 978-5045. ▨

Computer Algebra: Symbolic Manipulations Software

Andrzej J. Pindor
apindor@vm.utcs.utoronto.ca

When people talk about scientific software, they usually mean programs or packages that can do fancy numerical calculations, in other words, manipulate numbers. However, the numbers are manipulated always according to a certain prescription, i.e., a certain formula or a set of formulae. To arrive at such formulae, most of us must use pen and paper and do some head scratching. In fact, certain computer programs can help in this initial part of problem solving too, and they come into the realm of *computer algebra*. These programs can transform and simplify algebraic and trigonometric formulae, evaluate both definite and indefinite integrals symbolically, solve equations in terms of symbolic parameters, evaluate sums and products, solve differential equations, and so forth. As a bonus, they can translate mathematical formulae (entered by you or derived by the program) into one of the traditional computer languages, such as FORTRAN. The resulting code can then be output to a file and later incorporated into a numerical program.

This sounds great, but before you become convinced that at last you can have a slave mathematician on call to solve all your problems, consider the following situation: You would like to solve a complicated equation in terms of a number of parameters appearing in it; say, find unknown x in terms of parameters a and b, i.e., $x=f(a,b)$. What would constitute a "final result" for such a problem? x may not be expressible through elementary functions of a and b, and we then usually say that the equation cannot be solved in a closed form. But what do we count as elementary functions? Is the equation solved if we can express x through a gamma function of a and b? Even if this ambiguity is resolved, every (complicated) formula can be written in a number of different forms, and the one we settle for is very often a matter of personal taste. For example, you might you prefer the result as

$$\ln(\frac{a + b}{a^3 + b^3}) \quad \text{or as} \quad -\ln(a^2 - ab + b^2)$$

This is of course a trivial example, but you can easily imagine the number of possibilities arising in complicated cases. In addition, you can easily think of examples where trivial changes in the problem under consideration can drastically change the level of mathematical sophistication required to solve it.

Even in situations where there is a solution, it may be completely useless. For instance, it is well known that there exist closed-form solutions for roots of polynomials of degree 4 and less. Everyone learns the formulae for second order polynomials in high school, but trying to write down the formulae for the cubic and quartic polynomials leads quickly to so much tiresome algebra that no one in fact does it. However, a computer algebra program should be able to do it, shouldn't it? Of course it can, but an expression for a root of a quartic polynomial takes a few pages of algebraic symbols, so what is its use?

The above examples are intended to show that although computer algebra programs are sophisticated from the point of view of the code they represent, they also require sophistication on the part of the user and do not warrant high expectations. With this caveat, they may nevertheless be very useful, performing trivial tiresome tasks, preventing "slips of the pen," and saving us time otherwise spent searching through mathematical handbooks and tables of formulae. Here is a typical example of their use.

Suppose you want to find zeros of a complicated function of several variables. Numerical algorithms for such a problem work best if you can supply first or even second derivatives of the function with respect to all variables. You might hesitate to do so, however, because of the unlimited possibilities of error in taking derivatives and then coding the big resulting expressions, not to mention the mental pain of going through such a mundane task. However, a computer algebra program can do it all for you, saving you time and aggravation and also giving you the luxury of confidence that at least this part of your program is error free.

Even if your symbolic-manipulation requirements go beyond the capabilities mentioned above, it might be worthwhile to look into these packages, since they provide a means (i.e., a programming language) of extending their capabilities into other fields of mathematics; a lot of such extensions, supplied by other users, do in fact exist, for instance, in the field of high energy physics.

Until recently, such programs were available only on mainframe computers, due to their large requirements for computer resources. Their popularity was limited for a number of reasons. First, they use a lot of computer time and memory; thus, when run interactively, which makes the most sense for the type of tasks they perform, they are very expensive. Second, as with many mainframe products, they were not written with user friendliness in mind. Third, they were meant to be used from dumb terminals, which can only display text. As a result, mathematical formulae had to be represented in a very unnatural way. For instance, to find an integral of *sin(x)* over *x* from *a* to *b*, you may have to write something like

```
INTEGRATE(SIN(X),X,A,B)
```

As the power of smaller machines grew, the situation changed considerably. Now, not only are such programs becoming available on workstations and even microcomputers, where running cost is not an issue, but they are also becoming much more user friendly. For one thing, due to the improved graphics capabilities of these machines, the mathematical formulae, on input and output, finally look the way we expect.

The oldest, and probably most extensive of the computer algebra programs, MACSYMA, started life at MIT in the late 1960s and was then taken over by Symbolics Inc. Originally a mainframe product, it is now available on a number of workstations, including Apollo and SUN.

REDUCE is another symbolic-manipulation program dating from about the same period. It is available on a wide range of mainframes and recently also on the IBM PC. Tektronics uses REDUCE as an imbedded subsystem for its product MathScribe, which is basically a user-friendly envelope around REDUCE, with the addition of powerful 2D and 3D graphics and other enhancements.

More recent systems include Maple, developed since 1980 at the University of Waterloo. Originally a mainframe system, Maple has recently become available on microcomputers.

muMath (The Soft Warehouse, Honolulu, Hawaii) was designed in the early 1980s as a system for microcomputers. In an attempt to fit it on machines with small RAM, its capabilities were limited, and it has not been enhanced since then. muMath is reviewed in the preceding article.

The newest entry in the field is Mathematica. A product of Wolfram Research Inc., it is the first of these systems developed with the capabilities of a modern workstation in mind. It is available on SUN workstations, Macintosh microcomputers (the standard version runs on the Macintosh Plus, SE, or II, and some other systems.

UTCS is expecting to receive Mathematica (Mac II version) for a 30-day evaluation period. A report of this evaluation will appear in a later issue of ComputerNews. Meanwhile, a recent review of Mathematica can be found in the November 1988 issue of *MacUser*. ▨

MACSYMA is available from Symbolics Inc., 257 Vassar St., Cambridge, MA 02139

REDUCE comes from Rand Corporation, 1700 Main St., P.O. Box 2138, Santa Monica, CA 90406-2138.

Maple is available from WATCOM Products, 415 Phillip St., Waterloo, ON.

muMath is a product of Soft Warehouse Inc., 3615 Harding Avenue, Suite 505, Honolulu, Hawaii. It is available from many software vendors.

Mathematica comes from Wolfram Research Inc., P.O. Box 6059, Champaign, IL 61821.

SPSS^x Version 3.1

Joyce Bernstein
jbernst@vm.utcs.utoronto.ca

Now that you've gotten used to SPSS^x Version 3.0, a newer release is available for user test on CMS (UTORONTO and UTOREPAS). As usual, the new release operates exactly as the previous version, with the exception of several enhancements to existing procedures and facilities. The highlights of these new additions are discussed below; more information is available by issuing an appropriate INFO command from within SPSS^x.

A new procedure, RANK, enables you to produce ranks, normal scores, percentiles, and Savage (exponential) scores for numeric variables. The optional TIES subcommand on the RANK statement affords control over the treatment of tied ranks. The results of these computations are added to the active file as new variables. Names for newly created variables can be specified through use of the INTO subcommand. If INTO is not specified, RANK creates default names. These RANK created variables can be used for a number of nonparametric tests.

Several enhancements have been made to REGRESSION for Version 3.1. Two new keywords, COLLIN and SELECTION, have been added to the STATISTICS subcommand. COLLIN provides statistics for assessing collinearity, while SELECTION produces information for selecting a set of regressors. The CIN keyword on the CRITERIA subcommand affords control over the percentage used in confidence interval calculations.

Several statistics à la Belsley, Kuh, and Welsch for diagnosing influence and fit are now available for specification in the temporary variable lists on the CASEWISE subcommand. For example, a common worry in regression analysis is that an individual case or cases are unduly influencing the size of the model parameters. This situation can now be analyzed through the use of DFBETA on the CASEWISE variable list. Such a specification creates a casewise list of the change in each of the model's regression coefficients that would result from the deletion of the associated case from the analysis. Similarly, the use of DFFIT with the subcommand CASEWISE reports the change in the predicted values when individual cases are discarded. For more information regarding these and other new REGRESSION features, include the statement INFO REGRESSION in an SPSS^x program or interactive session.

A number of notable changes have also been made to the CROSSTABS procedure. The specification of KAPPA on the STATISTICS subcommand can be used to compute the kappa statistic for square contingency tables. Kappa is often used to measure agreement between two or more raters, or classifications of raters. Along with the value of kappa, the standard error of this statistic and the corresponding t-statistic is also reported. Among other additions to CROSSTABS is the facility to compute relative risk for 2X2 tables through use of the keyword RISK with the STATISTICS subcommand. Again, use INFO CROSSTABS for more information on the changes to this procedure.

Four new SET keywords appear in this release that control the routing of output. The command SET ERRORS TERMINAL sends error messages to the terminal only. Substituting the keyword LISTING for TERMINAL causes error messages to be sent to a listing file instead. Keywords BOTH or NONE can also be used either to send messages to both places or to suppress these messages completely. As you can confirm with the SPSS^x command SHOW ERRORS, the default route for error messages at UTCS is BOTH. The three other new SET keywords control resource utilization messages (MESSAGES), the echoing of commands read from a file (PRINTBACK), and the output of statistical results (RESULTS). The choice of routes described for the error messages applies to these cases as well.

You are reminded that the *SPSS^x User's Guide, 3rd Edition* is now available. This documents fully the features discussed above. Edition 2 of the *User's Guide* is still useful, especially if supplemented with a

continued on page 30

GAUSS News

Andrzej J. Pindor
apindor@vm.utcs.utoronto.ca

Aptech Systems recently announced a new version of GAUSS (this numerical and statistical software package was described in the April and May/June issues of *ComputerNews*) Version 2.0 is said to have many enhancements, on which we shall report once we have received it.

In the past two months, a few of you have inquired about forming a GAUSS users' group. Any GAUSS users interested in such a support group should contact me at 978-5045.

Please note that the support level for GAUSS at the UTCS Information Centre has been upgraded from Class C to B.

SPSS^x Version 3.1
continued from page 29.

printed copy of the INFO command output. The new manual is available for reference in the UTCS Information Centre, Room 201, 4 Bancroft Ave.

SPSS^x has also provided us with a listing of previously existing errors that have been fixed in this new release, along with a report on known errors not yet rectified. These lists are available to anyone interested. Access to this new product is available through the CMS command GETME SPSSX NEW. Please give Release 3.1 a workout and call 978-HELP to let us know of any problems.

WordPerfect 4.2 Course To Be Phased Out

Irene Rosiecki
rosiecki@vm.utcs.utoronto.ca

With the release of WordPerfect Version 5.0 , support for the older Version 4.2 will be diminishing.

Beginning in the new year, we will be reducing the number of WordPerfect 4.2 courses offered, and emphasis will be on Version 5.0. If you need to attend the Version 4.2 WordPerfect course, please call me at 978-4565.

Don't delay! Time is running out, and courses are filling up.

Minpack Library Available on CMS

Bill Fehlner
fehlner@vm.utcs.utoronto.ca

The Minpack-1 subroutine libraries were installed on UTORONTO in late October. Most of the functions available through Minpack can also be obtained from libraries such as IMSL or NAg. Minpack is made available for users who may have FORTRAN code written specifically to call Minpack routines. Unlike IMSL or NAg, the source code for Minpack is available.

Minpack is a set of FORTRAN subroutines designed to solve systems of nonlinear equations and nonlinear least squares problems. Both single- and double-precision versions of the subroutines are available.

How to access the package

The minidisk containing Minpack-1 is accessed with the command

```
getme minpack
```

Once this minidisk is accessed, Online Help is available with the command

```
help minpack
```

The Minpack minidisk contains source code, compiled versions in TXTLIBs, and machine readable documentation. To use these files, keep in mind:

1. Single-precision source code has file type FORTSGLE. Rename this to FORTRAN when copying onto your own minidisk.
2. Double-precision source code has file type FORTDOUB. Rename this to FORTRAN when copying onto your own minidisk.
3. The contents of the TXTLIBs were compiled using VS FORTRAN1.4.1 at PUT8704.
4. HELP MINPACK provides the details on linking one of the TXTLIBs with your compiled program.
5. Most subroutines have the same entry names in both single and double precision. If you insist on calling both single- and double-precision subroutines from the same program, you must recompile the single-precision ones, taking great care that the entry names are changed so that none of the entry names are the same as those in the double-precision library.

Documentation

As mentioned above, HELP MINPACK accesses Online Help once the Minpack minidisk has been accessed. Source code can be reviewed by browsing the source code files mentioned above. Details of the arguments and calling syntax for each subroutine are contained in MINPACKD LISTING and MINPACKS LISTING on the minidisk, for double- and single-precision routines respectively.

Additional documentation is provided in the first 50 pages of the manual, *User Guide for Minpack-1* by Jorge J. More, *et al.* This manual provides an overview of the algorithms and software in Minpack-1. It is available for reference at the UTCS Information Centre (SNAC Group).

Support level

Minpack has Class C support. UTCS has installed these subroutines from the version formerly available on MVS. It is recommended that users choose routines from a library with Class A whenever possible. Please report problems with Minpack to Advising Services at 978-HELP. No direct assistance is available, but problems will be referred to consultants. ▒

MINOS Available on CMS

Bill Fehlner
fehlner@vm.utcs.utoronto.ca

The MINOS package from the Department of Operations Research at Stanford University was installed on UTORONTO in early November. MINOS is a combined linear and nonlinear programming system. This FORTRAN-based system is designed to solve large-scale optimization problems that may have nonlinear terms in either the optimization function or the constraints or both.

How to access the package

The minidisk containing MINOS 5.1 is accessed with the command

getme minos

Once this minidisk is accessed, Online Help is available with the command

help minos

It is necessary to write some FORTRAN code to use the MINOS system. In addition, two other files must be provided to define a linear programming problem fully for the MINOS package. The SPECS file defines various run-time parameters such as the iteration limit. The MPS file specifies names for the constraints and variables and defines all the linear constraints and bounds. The data format of the MPS file is similar to that used in other mathematical programming systems.

Documentation

It is important to read at least Chapters 1, 2, and 8 of the *MINOS 5.1 User's Guide* by Bruce A. Murtagh and Michael A. Saunders before trying to use MINOS. This manual is available for reference in the UTCS Information Centre (SNAC Group).

As mentioned above, HELP MINOS accesses Online Help once the MINOS minidisk has been accessed.

Sample programs

Three examples, each including FORTRAN code, SPECS files, and MPS files, are provided on the MINOS minidisk. For instance, the test problem MANNE is a small example of an economic model. It has a nonlinear objective function, 10 nonlinear constraints, 10 linear constraints, and 30 variables. The starting point given in the MPS file is intentionally close to the optimum solution to make this an inexpensive test solution. Two other test programs on the MINOS minidisk are described in Chapter 7 of the MINOS *User's Guide*. Further examples of linear programming, unconstrained optimization, linear constrained optimization, and nonlinear constrained optimization are discussed in Chapter 8 of the *User's Guide*.

Support Level

MINOS has Class C support. UTCS has assisted in installing the package. Users are responsible for designing and implementing their problems in a format suitable for solution with MINOS. Please report problems to Advising Services at 978-HELP. No direct assistance is available, but problems will be referred to consultants. ▨

WordPerfect 1.0 for the Macintosh

Pat Hood
phood@vm.utcs.utoronto.ca

In January 1988, WordPerfect Corporation offered a prerelease (beta) version of WordPerfect for the Macintosh. After some months of user evaluation and feedback, many bug fixes and some design changes were made. This review is based on the full release version, which is now available.

Where to begin?

After opening a new document, you begin by setting the timed backup option in the WordPerfect defaults. This feature, apparently unique to WordPerfect software, saves the text in memory at timed intervals, providing a safety net in case of power failures, machine malfunctions, and software failures. The ability to set and save a number of program defaults (of which timed backup is one) is a wonderful feature; in addition to file backup options, you can choose a unit of measurement, define the error conditions that require a beep to alert the user, and specify default folders for saving and retrieving various types of documents. It is unfortunate that each of these functions has a separate menu and command box (there are five altogether) rather than there being a single command box for all.

The 4.2 model

Most of the functions found in WordPerfect 4.2 for the IBM PC have been incorporated in the Macintosh version; Sort, Math, and Line Draw are missing. The Macintosh product is best described as a transfer rather than a transformation. The software is easiest to use when you work in an IBM-like manner. Somewhere along the line the developers forgot that many Macintosh users have never used an IBM PC and do not intend to start! If the intention was to make the transition from PC to Macintosh easier for the user, then the product also fails because the Macintosh version is sufficiently different to make relearning WordPerfect necessary.

The screen

On the surface WordPerfect looks like any other piece of Macintosh software. It has the usual menu bars and scroll bars, although the horizontal scroll bar is extremely short because WordPerfect's status line uses the same space. Each document window has a Zoom box, which works only if you have changed the size of the window using the Size box.

Basic formatting commands (bold, italics, underline, outline, shadow, and plain text) are available from the status line, making character emphasis fast and simple. More complex formatting uses the menu commands or their keyboard equivalents.

Some screen tools that Macintosh users take for granted are text selection and the ruler. WordPerfect for the Macintosh does not have a selection bar like Microsoft Word; instead, the Select command lets you specify a portion of the text to be selected (sentence, paragraph, page, column, whole document). Text can also be selected by dragging across it with the mouse. WordPerfect does have a ruler. Its main functions are variable-width column definition and hyphenation zone definition. Also, the ruler is the only place where you can specify right justification of text. Tabs can be positioned on the ruler but cannot be dragged in order to change their position or to delete them. We found that tabs could not be positioned exactly on the quarter inch ruler markings. Even with the tab icon precisely aligned, the tab moved slightly left or right when we clicked to position it.

One interesting innovation in WordPerfect is a command that allows you to hide the menu bar, title bar, and right scroll bar, replacing them with lines of your document. When you move to the top of window with the mouse, the menu bar is displayed again. The title bar and vertical scroll bar do not reappear until you turn the function off. Without a vertical scroll bar it is difficult, though not impossible, to move around in the document.

The Hierarchical Menu System

WordPerfect takes advantage of Apple's new hierarchical menu structure wherein submenus of commands are displayed directly from the main menu. Touted as a great new feature of WordPerfect, the hierarchical menus are not a big advantage; however, they do dilute the frustration of proceeding through multiple levels to get to a command. The multiple levels are a carryover from the IBM version of WordPerfect. They are just as cumbersome as those in Version 5.0 on the IBM. In WordPerfect on the Macintosh, many of the commands that use the hierarchical menu system do not have keyboard equivalents, forcing the cumbersome entry system on the user.

The keyboard

One way around the inefficient multilevel commands is to map functions to key combinations involving the Enter key plus a character or <Enter><Enter>plus a character. (For users with Extended keyboards, WordPerfect also provides Function key mapping like that on the PC.) If the Enter key were more conveniently located (i.e., like the Ctrl or Shift keys of the IBM keyboard) this would be a great function. It would quicken access to a particular function buried deep in the menu hierarchy.

The WordPerfect manual has well organized introductory information, especially concerning the various Macintosh keyboards. It goes into detail about substituting keystrokes and functions in the absence of an Esc key, cursor keys, or a numeric keypad. An appendix illustrates how the Extended keyboard function keys may be used and provides information for obtaining a template that shows the key mapping. The four-level function key mapping and template are other carryovers from the IBM PC world and do not fit as well in a mouse-oriented environment. Some of the mapping is identical to that for the IBM PC, but for PC users, there are keys that will have to be relearned.

Ways to manage files

In keeping with the spirit of the Macintosh, WordPerfect has Open, Save, and Close commands on the File menu. In addition, it has a Retrieve command on the File menu, and under File Management it has both an Open button and a Retrieve button. Open and Retrieve are different functions (both open files, Retrieve is meant for merging one document with another), but new users may be confused since the effects are interchangeable in some situations. The File Management command is a nice idea but may not be necessary. The Macintosh desktop is so much easier to manage than the PC file system that this carryover from the PC could be eliminated without loss of functionality to the user.

Functions, functions everywhere

One of the most positive aspects of WordPerfect is the breadth of functions. It is not just a matter of having more functions than other word-processing packages. WordPerfect's basic functions often provide more flexibility than other applications. Examples include special print options for printing every other page and determining when the cover page will print, and Open command box buttons to limit which files are listed in the scroll boxes.

One important feature of WordPerfect is automatic repagination (Microsoft Word does not have this ability). The status line always indicates the page and line position of the cursor. It can also display the absolute position on the page of the cursor for precise positioning of graphics and text.

Unlike Microsoft Word, WordPerfect shows true column positioning on the screen (instead of in a special preview screen). Columns can be defined using a menu command or, more quickly and easily, using the Ruler. Up to 24 columns can be placed side-by-side on a page. Parallel block columns can also be created. These are an upscale version of Word's side-by-side paragraphs.

WordPerfect's Speller is much better than Microsoft Word's. It automatically suggests replacements for misspellings under the categories *typos* and *phonetic*. You can choose to ignore a word throughout a document or only skip it once; Word only allows you to ignore words. You can limit spell checking to a selection or page. You can ask for a word count without having to check spelling as well. The only drawback to using the speller is that for longer documents the change between the document and speller screens takes a few seconds; it is disconcerting to see your document disappear and then just as quickly reappear.

The Thesaurus is almost identical to that found in the PC version. This is not surprising since the thesaurus is the most window-like portion of WordPerfect on the PC.

Merge, for creating form letters, is slightly more difficult in WordPerfect than in Microsoft Word because of the field-definition structure used. It is just as flexible however, providing *if...else* statements, keyboard entry on merging, and macro inclusion. Index and table of contents generation is also slightly more difficult in WordPerfect, but there is more flexibility in the style and placement of these lists within a document.

As in Microsoft Word, WordPerfect allows you to place graphics within a document. Taking the idea another step in the right direction, WordPerfect links a graphic to a paragraph, making it easy to place text next to a graphic. When the document is reformatted, the graphic stays with the paragraph, moving between pages as required. Graphics can be sized to fit into columns or with a specific piece of text. It is in the more complex functions such as these that WordPerfect truly shines.

Reluctant WYSIWYG

WordPerfect for the Macintosh feels like a product reluctant to move into a completely "What You See Is What You Get" world. Font sizes and families are restricted. Justified text cannot be edited accurately. (WordPerfect provides a command to show justified text unjustified on the screen while you are editing.) Formatting text places hidden codes in the document — codes that can be viewed and edited using a special screen only. The codes are another carryover from WordPerfect for the PC. While they make sense in an environment where little text formatting can be displayed on the screen, on the Macintosh they are a hindrance.

As in WordPerfect for the PC, formatting codes multiply at astronomic rates often causing formatting havoc. Why bother with codes when what you see on the screen is supposed to be what is going to print? In our view, the software should handle all the complexities of giving you what you ask for, without making you worry about how it does it. WordPerfect's reliance on hidden codes is an unnecessary complication in a word-processing package that is already awkward and difficult at times.

In conclusion

WordPerfect 1.0 suffers from a split personality. It has lots of functions but, in design, many are better suited to the PC environment. Others, like timed backup, automatic repagination, and graphics linked to paragraphs are great innovations for the Macintosh word-processing environment, and Microsoft should take note. Complex features such as the spelling checker and merge are well-designed and easy to use. However, the simple functions that you would use everyday are often awkward, confusing, and/or time-consuming. Because it can read WordPerfect 4.2 (IBM) and Microsoft Word (Mac) files, WordPerfect on the Macintosh will allow PCs and Macs to live harmoniously, sharing the same word-processing package, and we expect the package to occupy this niche. Until it differs significantly, in features or price, from the many other Macintosh word-processing packages, WordPerfect on the Macintosh still has to prove itself to other users. ▨

VM/CMS Language Processor Upgrade

Herb Kugel
herb@vm.utcs.utoronto.ca

Production processors on both the UTORONTO and UTOREPAS computers will change in early December. PL/I 5.1 and VS FORTRAN 1.4.1 will receive maintenance upgrading on both machines. The following will take place on UTORONTO only:

WATFOR-77 2.5 will become the production WATFOR-77 Compiler, replacing WATFOR-77 1.5.

Waterloo C 3.0 will become the production Waterloo C Compiler, replacing Waterloo C 1.3.

VMSORT (the production Sort that is called by SAS, SPSS, and PL/I) will be augmented, Release 2.1 replacing Release 2.0.

Please watch Hot News for additional information.

ProComm Communications Program Seminar

January 18, 1989
9:00 - 11:00 a.m.

ProComm is a general purpose communications program currently available to all U of T faculty, staff, and students. It is the resident communications program in the IBM Education Facility.

A seminar has been scheduled to introduce ProComm and its various features: terminal emulation (a number of popular ones will be reviewed); the dialing directory; file transfer protocols, such as Kermit and XMODEM; configuration and modem parameter settings; and the DOS gateway, which allows you to execute DOS commands and other programs while you are still in ProComm.

A course fee is attached to this seminar. Copies of ProComm are made available at no charge to members of the University, but you must bring your own diskettes (two 5.25" or one 3.5").

To reserve a place in this seminar, please contact Irene Rosiecki at 978-4565.

Free NetNorth (Electronic Mail) Demonstration

January 18, 1989
11:00 - 12:30 noon

Anyone interested in sending documents to colleagues or friends reliably, quickly, and inexpensively will not want to miss this demonstration. We will introduce you to the NetNorth/BITNET/EARN communications network, which is accessible from the CMS system. This network connects more than a thousand universities and research institutes in Europe, Asia, and North America. Sending documents or files over the network is very easy to learn.

The demonstration will last just over an hour. Included will be how to send mail and files to another institution, and how to talk directly to someone on the network. ***Please note this is a demonstration only.***

Space is limited. If you would like to attend one of these sessions, please contact Irene Rosiecki at 978-4565 to reserve a place.

Documents at UTCS

All vendor-produced documentation may be ordered through the UTCS Information Office, Room 201, 4 Bancroft Avenue. UTCS documents and selected vendor documents are also stocked there for purchase. For more information on prices, call Dale Wright at 978-4990. Whenever feasible, documents are made available online, and users are encouraged to print their own copies. On the VM machine, a special command has been provided for this purpose.

To print online documents on:

IBM VM/CMS	***GP UNIX***
Type "help document"	Use *ipr* command

The following examples relate to the table below. The document to be printed is the UTCS Guide to Census Tapes, called CENSUS LISTING on CMS and /usr/doc/utcs/census on GP UNIX. <RETURN> means RETURN or ENTER key.

CMS:

```
document <RETURN> <RETURN>
laser census listing (forms yxle <RETURN>
```

GP UNIX:

```
nohup ipr -8p -fyxle /usr/doc/utcs/census & <RETURN>
```

UTCS Documents in hardcopy

New:

Previously announced:

- Academic's Guide to Microcomputer Systems
- Character Sets for Highspeed Printing
- UTCS GUIDE TO PRODUCTS AND SERVICES: Introduction
- UTCS GUIDE TO PRODUCTS AND SERVICES: Basics
- UTCS GUIDE TO PRODUCTS AND SERVICES: Accounting
- UTCS Catalogue: Numerical Products
- UTCS Catalogue: Statistics Products
- UTCS Catalogue: Text Products
- UTCS Guide to Census Tapes
- UTCS Guide to Kermit
- UTCS Guide to Micro File Transfer Service
- UTCS Guide to Text Processing on GP UNIX

UTCS Documents online

	CMS LISTING	GP UNIX	Forms Code
New:			
Previously announced:			
UTCS GUIDE TO PRODUCTS AND SERVICES: Introduction	INTRO	/usr/doc/utcs/intro	*
UTCS GUIDE TO PRODUCTS AND SERVICES: Basics	BASICS	/usr/doc/utcs/basics	*
UTCS Catalogue: Access, Part 1	ACCESS1	/usr/doc/utcs/access1	*
UTCS Catalogue: Access, Part 2		/usr/doc/utcs/access2	*
UTCS Catalogue: Numerical Products	NUMCAT		*
Product-Function List	NUMLIST		BXQD
UTCS Catalogue: Statistics Products	STATCAT		*
Product-Function List	STATLIST		BXQD
UTCS Catalogue: Text Products	TEXTCAT	/usr/doc/utcs/textcat	*
UTCS Guide to BMDP	BMDP		*
UTCS Guide to Census Tapes	CENSUS	/usr/doc/utcs/census	*
Recipes for Using Tapes on VM/CMS	QUIKTAPE		*
UTCS Guide to FORTRAN on VM/CMS	FORTRAN		*
UTCS Guide to GP UNIX		/usr/doc/utcs/gpunix	*
UTCS Guide to Highspeed Printing	PRINT	/usr/doc/utcs/print	*

* *Recommended forms code for the 8700 is YXLE unless otherwise stated.*
† *Use the 87set command. Type "help 87set".*

Documents continued

UTCS Documents online continued

	CMS LISTING	GP UNIX	Forms Code
UTCS Guide to Kermit	KERMIT	/usr/doc/utcs/kermit	*
UTCS Guide to Micro Laser Printing Service	MICRLASR		*
UTCS Guide to SAS	SAS		*
UTCS Guide to SCRIBE	SCRIBE		*
(For printing instructions, see page 1 of document)			
UTCS Guide to Series/1 Terminal Use	SERIES		*
3270 Emulation:			
Cybernex APL100 and Volker Craig VC404	CYBER		*
DM1520	DM1520		*
Hardcopy Terminal	HARDCOPY		*
IBM 3101	IBM3101		*
IBM PC Kermit	PCKERMIT		*
SOROC IQ	SOROC		*
VT52	VT52		*
VT100	VT100		*
UTCS Guide to SPSS	SPSS		*
UTCS Guide to Text Processing on GP UNIX		/usr/doc/utcs/text	YXOG
UTCS Guide to TFW.MAK in SCRIBE	TFWMAK		*
UTCS Guide to VM/CMS	CMSGD		*
UTCS Guide to WSCRIPT	WSCRIPT		*
UTCS Rate Schedule (Commercial)	RATESCOM		†
UTCS Rate Schedule (External)	RATESEXT		†
UTCS Rate Schedule (Internal)	RATESINT		†

Other Documentation online

	CMS LISTING	GP UNIX	Forms Code
Kermit			
Kermit Protocol Manual	KERMITPM		YXOE
Kermit User Guide	KERMITUG		YXOF
ProComm			
ProComm Reference Manual	PROCOMM		YXOE
VM Tape Documents			
TapeMap User's Guide and Reference	TAPEMAP		*
VMTAPE Messages and Codes	VMTMSGS		YXOE
VMTAPE User's Guide	VMTUSER		YXOE
VMSECURE			
VMSECURE User's Guide	VMSECURE		YXOE
Waterloo SCRIPT			
Reference Manual	WSCRPTRF		YXLC
User's Guide	WSCRPTUG		YXLC
Using SCRIPT in MVS/TSO	WSCRPTSO		YXLC
Waterloo SCRIPT and a PostScript Printer	WSCRPTPS		YXLC
Formula Processor Summary	WSCRPTFP		YXLC
GML User's Guide	GMLUG		YXLC
GML Reference Summary	GMLRFSUM		YXLC
OBS WYLBUR			
Messages and Codes			YXLD
Qwik Files			YXLD
Reference Summary			YXLG
Reference Manual			YXLD
7.0 Release Guide			YXLC
User Guide			YXLD
Electronic Mail			
A User's Guide to Electronic MAIL	MAILBOOK		YXOC

* *Recommended forms code for the 8700 is YXLE unless otherwise stated.*
† *Use the 87set command. Type "help 87set".*

Consulting and Enquiries

Applications Support & Advising Supervisor	Alex Nishri	BC214	978-7109
External Marketing Consultant	Ihor Prociuk	BC217	978-6875
Erindale College	Joe Lim	ER2035	828-5311
Information Office	Dale Wright	BC201	978-4990
Account & Access Code Enquiries (U of T)	tba	BC101B	978-8703
Account & Access Code Enquiries (External)	Sylvia May	BC101B	978-7148
IBM PC Maintenance	Kam Mark	BC103	978-5050
Tape Library (Academic Services)	Susan Kovago	MP368	978-7319
Tape Library (Administrative Services)	Miranda Fong	MP368	978-6693
Terminal Service	Rosi Tseu	BC105	978-7087
UTCS Noncredit Short Courses	Irene Rosiecki	BC217	978-4565

Consulting & Advising Services:	978-HELP
CMS, and GP UNIX userids:	ADVISOR
CMS Userid for mail problems:	POSTMSTR
System Status Enquiries (GP UNIX)	978-4318
System Status Enquiries (IBM)	978-7393
Interactive Services 300 (bps)	978-6200
Interactive Services 1200 (bps)	978-3959
Interactive Services 2400 (bps)	978-7239
Datapac	4320 0056
Telnet	0302043200056
Tymnet	<backspace>DPAC;302043200056

UTCS Directory

Director:	Dr. Warren Jackson	BC118	978-8948	wcj@vm.utcs.utoronto.ca
Associate Director:	Eugene Siciunas	BC116	978-5058	eugene@vm.utcs.utoronto.ca
Managers:				
Communications &Technical Support	Norman Housley	BC121B	978-4967	norman@vm.utcs.utoronto.ca
Information Centre	Don Gibson	BC217	978-7331	don@vm.utcs.utoronto.ca
Internal Systems Support	Ron Vander Kraats	BC121A	978-4428	rvk@vm.utcs.utoronto.ca
Operations Support	Dr. Bob Chambers	MP350	978-7092	
Systems Support	Bill Lauriston	MP331	978-3579	bill@vm.utcs.utoronto.ca

Committees on Computing

Committee on Administrative Computing	Chair	Janice Oliver	978-4322
UTCS Board	Chair	Prof J.F. Keffer	978-4984
Supercomputer Users' Group at U of T	Chair	Prof. J.M. Ting	978-4971
Research Board Standing Committee on Computing	Chair	Prof. C.C. Gotlieb	978-2986

Legend

BC = Bancroft Building
ER = Erindale
MP = McLehnan Physical Labs

* NetNorth BITNET EARN

UTCS Terminal and Advising Sites

Names and Locations

Central Advising Office (CAO), 978-HELP
Engineering Annex (EA, CDF), 11 King's College Road, Rooms 103, 107, 107B, 201, 203
Erindale College (Erin), 828-5339, Mississauga Road, Erindale Campus, Rooms 2037, 2039A-B-C, 2045
Robarts, Robarts Library, 130 St. George St., Room 1061A
Sidney Smith (Sidney), 100 St. George St., Rooms 1071, 2105
St.Michael's College, 121 St. Joseph St., Room 107
Trinity College, 6 Hoskins Ave., Room 024
Victoria University, 73 Queens Park Cres., Room 208

Access Hours

Sites	Hours of Access				Restrictions*	Advising
	Mon-Thurs	Fri	Sat	Sun		
CAO	10:00-18:00	10:00-18:00	closed	closed	Research	978-HELP
CDF	24 hrs	24 hrs	24 hrs	24 hrs	Undergrads	No
EA	24 hrs	24 hrs	24 hrs	24 hrs	None	978-HELP for Research
Erin (2039)Δ	9:00-17:00	9:00-17:00	12:00-16:00	12:00-16:00	Research	Rm 2005
(2045-46-47)	24 hrs	24 hrs	24 hrs	24 hrs	Undergrads	Rm 2005
(235)†	8:00-22:00	8:00-17:00	closed	closed	None	Rm 2046
Robarts	8:30-24:00	8:30-24:00	9:00-22:00	13:00-22:00	None	978-HELP for Research
Sidney	7:00-24:00	7:00-24:00	7:00-24:00	7:00-24:00	None	978-HELP for Research
St.Michael's	24 hrs	24 hrs	24 hrs	24 hrs	None	978-HELP for Research
Trinity	8:00-22:00	8:00-22:00	8:00-22:00	8:00-22:00	None	978-HELP for Research
Victoria	8:30-23:00	8:30-23:00	closed	closed	None	978-HELP for Research

Sites	Advising Hours
CAO	Monday through Friday, 10:00 - 18:00
Erin	Monday through Friday, 09:00 - 22:00

Legend

* Research includes graduates, faculty, staff.
\ Key access available.
† Access restricted to building hours.

Terminal and Printer Access

Sites	PACX	Network Terminal Server	Printers	CDF/PC Terminals
CDF	N	N	Y	N
EA	N	Y	Y	N
Erin	Y	Y	Y	N
Robarts	Y	N	Y	Y
Sidney	N	Y	Y	Y
St.Michael's	Y	N	Y	Y
Trinity	Y	N	Y	Y
Victoria	Y	N	Y*	Y

(Y=yes, N=no)
* CDF/PC only

UTCS Services

UTCS Centrally Owned and Managed

Systems

IBM VM/CMS

- General Purpose Timesharing; access to NetNorth, BITNET, and EARN networks; access to CRAY X-MP/22
- 4381 P02 processor, 16 megabytes of memory
- CP operating system, VM/SP HPO R4.2
- CMS timesharing system, VM/SP Release 4
- RSCS spooling system, Release 3
- CRAY station Release 4

GP UNIX
Technical Assistance: 978-8853

- General Purpose Timesharing under SUN UNIX 3.2
- SUN 3/280, 8 megabytes of memory
- access to Linotronic typesetter
- offers a sophisticated and powerful environment for text processing
- offers a sophisticated programming environment suitable for commercial software development or testing
- full access to USENET, an electronic technical information exchange facility
- offers excellent electronic mail facilities and ability to send or receive mail from most computer networks worldwide, such as UUCPNET, ARPANET, NetNorth, BITNET, CSNET, CDNNET

Services

Communications & Technical Support
Primary Phones: 978-3787, 978-4967

- Communications Group provides communications systems, terminals, modems, data channels: consulting and installation.
- Field Service Group installs and maintains communications and computer systems, particularly IBM PCs, Micro Vaxes, and Macintoshes, on a contract basis or on a cost-per-call basis.
- provides consulting on computer systems technology and installs computer systems
- provides access between the IBM systems and machines using UNIX, VMS and other operating systems. Ethernet, Pronet, and IBM TRN technologies are used over various transmission media including optical fibre. More basic communications techniques are also used for moderate speed links.
- provides consulting on local area networking and installs LANS
- provides access to NetNorth (BITNET), the North American Universities Network; ONET, the Ontario Regional Network; and USENET, the UNIX networking fraternity
- provides gateway to the US Internet
- will provide a communications solution to department needs on a contractual basis

Information Centre
Primary Phone: 978-HELP

- Provides assistance in use of electronic messaging, including use of Local Area Networks (LANS); NetNorth/BITNET/EARN, the world Universities network; CDNNET, the Canadian X.400 network; ARPANET; CSNET; USENET; and other international connections.
- provides advising, consulting and documentation on:
  - command languages, including CMS and UNIX
  - high level languages, including FORTRAN, and PL/I
  - packages and libraries, including SAS, SPSS-X, BMDP, IMSL, and NAg
  - editors and formatters, including XEDIT, ed, nroff/troff
- provides general micro support
  - selection consulting for hardware and software
  - Micro Lab for evaluation of hardware and software
  - advice on University discounts
  - media conversion and data transfer
  - operates UTCS Microcomputer Bulletin Board System
  - offers Micro Laser Printing Service
  - offers 35mm slide production service using Polaroid Palette
  - administers Local Area Network of PCs for Education Facility
  - administers Macintosh Education Facility
- high quality typesetting
- installs and maintains application packages
- provides short courses and seminars on the more popular services and software packages

> *The principal mandate of UTCS is to plan, implement, and operate facilities and "common-carrier" networks, and to plan and support divisional, departmental or project computer facilities as requested.*

Facilities Managed by UTCS

Administrative Computing

- administrative IMS/VS, DB/DC, DB2, Batch, and TS0
- 4381-R14 processor, 32 megabytes of memory
- MVS operating system

Computer Disciplines Facility/UNIX

SUN 3/280s, 8 megabytes of memory (zero.cdf)
- Computer Science interactive access
- SUN UNIX 3.5

SUN 3/280s, 16 megabytes of memory (one.cdf)
- Computer Science interactive access
- SUN UNIX 3.5

Computer Disciplines Facility/PC

- 97 TI Professional Computers connected in a Local Area Network
- introductory Computer Science instruction
- MS-DOS with Turing environment

Erindale College Systems

VAX 8200, 8 megabytes of memory
- instructional access using VMS
- research access

VAX-11/750, 5 megabytes of memory
- instructional access using UNIX (Berkeley UNIX 4.3BSD)

Institutional Relations System

VAX-11/750, 8 megabytes of memory
- database services to the owner departments using VMS

EPAS Facility

- 4361-5 processor, 16 megabytes of memory
- general VM/CMS services to the owner departments

Ontario Centre for Large Scale Computation (OCLSC)

- Cray Research Inc. X-MP/24
- 2 processors, 4 megawords main memory
- solid state disk (SSD) with 64 megawords of storage
- COS 1.16 operating system
- VAX8350 + MicroVAX II and VAX750 provide VMS front-end services
- VM front-end services provided by UTCS 4381-P02

UTCS Software List

SOFTWARE LIST
for
MAINFRAME
COMPUTERS
VM/CMS, UNIX

Each programming language or package available at UTCS has a support level. In general, Class A Products receive full support, Class B limited support (as time and resources allow), Class C no active support.

For complete details of UTCS product support policy, see the UTCS GUIDE TO PRODUCTS AND SERVICES: Introduction.

Package/Compiler	System	Support	Version
Communications			
Kermit	VM/CMS	A	---
Kermit	UNIX	B	4C(058)
FTS	VM/CMS	A	---
MAIL/MAILBOOK	VM/CMS	A	86.359
MICROLASER	VM/CMS	A	---
macget/macput	UNIX	C	---
netcopy, netexec, to (NetNorth Tools)	UNIX	B	---
xmodem	UNIX	C	---
ftp,rsh,rlogin,telnet,rcp (Access via TCP/IP to most of the networked machines on all three campuses)	UNIX	B	---
rn/Pnews	UNIX	B	---
mail,mh,Mail	UNIX	-	---
Graphics			
PLOTBASIC (FORTRAN)	VM/CMS	C	---
SAS/GRAPH	VM/CMS	A	5.16
Languages			
ASSEMBLER(F)	VM/CMS	C	---
Concurrent Euclid	VM/CMS	C	1.9
FORTUTILS	VM/CMS	A	2.2
PL/I OPT Compiler	VM/CMS	B	1.5.1
REXX	VM/CMS	A	---
SNOBOL 4	VM/CMS	C	3.5
Turing	VM/CMS	C	---
VS FORTRAN	VM/CMS	A	1.4.1
VS Pascal	VM/CMS	C	2.2
Waterloo C	VM/CMS	C	1.3
WATFIV	VM/CMS	C	V2L0
WATFOR-77	VM/CMS	A	1.3
cc	UNIX	A	---
f77	UNIX	C	---
ratfor	UNIX	C	BSD std.
ttc	UNIX	C	---
pc	UNIX	C	BSD std.
yacc	UNIX	C	---
awk	UNIX	C	---
adb,dbx	UNIX	C	BSD std.
Numerical			
IMSL	VM/CMS	A	9.2,10.0
LINPACK	VM/CMS	C	---
NAg	VM/CMS	A	Mark 11
Statistical			
BMDP	VM/CMS	A	Level 85, Level 87
Minitab	VM/CMS	B	5.1
NAg	VM/CMS	A	Mark 11
NTSYS	VM/CMS	C	---
SAS	VM/CMS	A	5.16
SPSS^x	VM/CMS	A	2.2

Software List continued

Package/Compiler	System	Support	Version
Text Processing			
SCRIBE	VM/CMS	C	4(1400)-1
Waterloo SCRIPT	VM/CMS	B	86.1
XEDIT	VM/CMS	A	---
8700 Pseudotypesetting	VM/CMS	C	n/a
MICROLASER	VM/CMS	A	---
ed	UNIX	A	---
vi	UNIX	C	---
cmacs	UNIX	C	---
spell	UNIX	C	---
diction/explain	UNIX	C	---
eqn	UNIX	B	---
tbl	UNIX	B	---
pic	UNIX	C	---
troff	UNIX	B	---
nroff	UNIX	B	---
TeX	UNIX	C	---
Utilities			
FORTUTILS	VM/CMS	A	2.2
TPRINT	VM/CMS	C	---
crypt	UNIX	A	BSD std.
compress	UNIX	C	BSD std.
csh	UNIX	B	BSD std.
diff	UNIX	A	BSD std.
grep	UNIX	A	BSD std.
ipr	UNIX	B	---
learn	UNIX	C	BSD std.
patch	UNIX	C	---
sed	UNIX	A	BSD std.
sort	UNIX	A	BSD std.
sh	UNIX	A	BSD std.
tar	UNIX	A	BSD std.
uniq	UNIX	A	BSD std.

Software List continued

SOFTWARE LIST for MICROCOMPUTERS

The following are microcomputer software packages for which UTCS can provide either support or some information. As with mainframe products, micro packages receive full support (Class A), limited support (Class B), or no active support (Class C). The specific criteria differ from those for mainframe classes and are outlined in the Academic's Guide to Microcomputer Systems.

Package/Compiler	Version	Environment	Support
Application Development Tools			
BASIC			
GW BASIC	—	MS-DOS	B
C			
Microsoft C	5.1	MS-DOS	B
MPW C	2.0.2	Macintosh	B
WATCOM C & Express C	6.0	MS-DOS	B
FORTRAN			
IBM FORTRAN/2	1.0	MS-DOS 3.3 & OS/2	B
Language Systems FORTRAN	1.0	Macintosh	C
Microsoft FORTRAN	4.0	MS-DOS	B
WATCOM WATFOR-77	3.0	MS-DOS	A
Pascal			
MPW Pascal	2.0.2	Macintosh	C
TURBO Pascal	—	MS-DOS	B
Tools			
ResEdit	1.2d0	Macintosh	B
HyperCard	1.2.1	Macintosh	A
Macintosh Programmer's Workshop	2.0.2	Macintosh	B
WATCOM GKS	1.3	MS-DOS	B
Communications			
File Transfer & Terminal Emulators			
Kermit	0.8(35)	Macintosh	A
Kermit	2.30	MS-DOS	A
Kermit	—	Apple II DOS	C
Kermit	—	CP/M	C
Lap-Link DeskLink	1.0	MS-DOS	B
Lap-Link Mac	1.2	Macintosh	B
Lap-Link PC	2.16	MS-DOS	B
MacTerminal	2.2	Macintosh	B
PC TALK3	—	MS-DOS	C
PC PLOT III	—	MS-DOS	C
ProComm	TU 2.4.2	MS-DOS	A
Red Ryder	10.3	Macintosh	B
Network Systems			
AppleShare	2.0.1	Macintosh	B
Novell Advanced Netware	2.0a	MS-DOS	B
TOPS	2.0	Macintosh	B
TOPS	2.0	MS-DOS	B
Network System Utilities			
Inter•Poll	1.0	Macintosh	C
Database Managers			
4th Dimension	Int. 1.03	Macintosh	B
AskSam	3.0	MS-DOS	B
Clipper	S87	MS-DOS	B
DATAEASE	2.5	MS-DOS	B
dBASE II	—	CP/M	B
dBASE III Plus	1.1	MS-DOS	A
FoxBASE	2.14u	MS-DOS	C
Notebook II	2.03	MS-DOS	C
REFLEX	—	MS-DOS	B
ViewFinder III	—	MS-DOS	B
ZyINDEX	—	MS-DOS	B

Software List continued

Package/Compiler	Version	Environment	Support
Desktop Publishing			
PageMaker	3.0	Macintosh	A
PageMaker	3.0	MS-DOS	A
Ventura	1.1	MS-DOS	B
XPress	1.0	Macintosh	C
Electronic Mail			
Microsoft Mail	1.36	Macintosh	B
QuickMail	1.0	Macintosh	B
Graphics			
Charting			
35mm Express	4.02	MS-DOS	B
Cricket Graph	1.2	Macintosh	C
Freelance Plus	2.01	MS-DOS	B
Lotus 1-2-3	2.01a	MS-DOS	B
Microsoft Excel	1.5	Macintosh	B
Microsoft Excel	2.0	MS-DOS	B
SigmaPlot	—	MS-DOS	C
Draw and CAD			
Aldus Freehand	1.0	Macintosh	B
Cricket Draw	1.1	Macintosh	C
Generic CADD	3.0	MS-DOS	C
MiniCad	4.05	Macintosh	C
Illustrator	1.0	Macintosh	B
Illustrator88	1.6	Macintosh	B
MacDraft	1.2a	Macintosh	C
MacDraw	1.95	Macintosh	A
MacDraw II	1.0v1	Macintosh	B
Windows Draw	1.02	MS-DOS	C
Image and Scanning			
ImageStudio	1.05	Macintosh	B
Publish Pac	2.0	Macintosh	B
Publish Pac	2.0	MS-DOS	B
Text Pac	2.0	MS-DOS	B
Paint			
DR HALO	1.23	MS-DOS	C
MacPaint	2.0	Macintosh	A
Microsoft Paintbrush	2.0	MS-DOS	C
PC Paintbrush	2.5	MS-DOS	C
SuperPaint	1.1	Macintosh	B
Windows Paint	2.03	MS-DOS	C
Plotting Subroutines			
WATCOM GKS	1.3	MS-DOS	B
Numerical			
Eispack	n/a	MS-DOS	C
Eureka	n/a	MS-DOS	C
GAUSS	1.49B	MS-DOS	C
Linpack	n/a	MS-DOS	C
MathCAD	2.0	MS-DOS	C
muMATH-83	4.12	MS-DOS	C
Operating Systems			
IBM PC, PS/2, and compatibles			
IBM OS/2	1.0	n/a	B
IBM PC-DOS	3.3	n/a	B
Microsoft MS-DOS (incl IBM PC-DOS)	3.2	n/a	A

Software List continued

Package/Compiler	Version	Environment	Support
Microsoft Windows	2.03	n/a	A
Microsoft Windows 386	2.03	n/a	C
Macintosh II, Macintosh SE, and Macintosh Plus			
Finder	6.1	n/a	A
MultiFinder	6.0.1	n/a	A
System	6.0.2	n/a	A
Apple II			
Apple II DOS	3.3	n/a	C
CP/M	2.20	n/a	C
ProDos	1.0	n/a	C
Sound			
SoundWave	1.0	Macintosh	B
MacRecorder	1.0	Macintosh	B
Spreadsheet			
APPLEWORKS	—	MS-DOS	C
HAL	1.0	MS-DOS	B
Lotus 1-2-3	2.01a	MS-DOS	B
Microsoft Excel	1.5	Macintosh	B
Microsoft Excel	2.0	MS-DOS	B
Multiplan	1.5	Macintosh	C
Statistical			
MACSPIN	1.5	Macintosh	C
Minitab	82.1.1	MS-DOS	C
PowerStat	1.04	MS-DOS	C
SAS/PC	6.03	MS-DOS	A
SPSS/PC+	2.0	MS-DOS	A
STATPRO	2.0c	MS-DOS	C
STATWORKS	1.2	Macintosh	B
SYSTAT	3.2	Macintosh	B
SYSTAT	3.2	MS-DOS	B
Word Processing			
FinalWord II	2.0	MS-DOS	B
MacWrite	5.01	Macintosh	B
Manuscript	1.0	MS-DOS	C
Microsoft Word	3.01	Macintosh	A
Microsoft Word	4.0	MS-DOS	A
Nota Bene	3.0	MS-DOS	B
Nota Bene SLS	3.0	MS-DOS	B
PC-TeX	2.1	MS-DOS	B
Sprint	1.0	MS-DOS	B
T^3	—	MS-DOS	C
Volkswriter Scientific	—	MS-DOS	C
WordPerfect	4.2 & 5.0	MS-DOS	A
WordPerfect	1.0	Macintosh	C
WordStar	3.3	MS-DOS	B
Utilities			
Apple Turnover	—	MS-DOS	A
FASTBACK	—	MS-DOS	B
Norton Utilities	4.0	MS-DOS	A
Copy II Disk	7.1	Macintosh	B
MacZap	5.1	Macintosh	B
Sidekick	—	MS-DOS	B
SuperKey	—	MS-DOS	C
TURBO Lightning	—	MS-DOS	C
XenoDisk	1.41	MS-DOS	A

Cut Here

Annual *ComputerNews* Subscription Renewal Form

RENEW YOUR SUBSCRIPTION !

If you wish to receive *ComputerNews* for another year, please fill out the form below and return it promptly. Otherwise, your subscription will expire after Issue #251, November/December 1988.

NAME, and ADDRESS in full
(or attach your mailing label, indicating any changes)

____

____

____

____

☐ Check here if you can receive *ComputerNews* by campus mail.

— — — — — — — — — — — — — — — — FOLD HERE — — — — — — — — — — — — — — — —

Return to:

ComputerNews
University of Toronto Computing Services
4 Bancroft Avenue, Room 217
Toronto, Ontario
M5S 1A1

Cut Here

UTCS *ComputerNews* Subscription Request Form

If your wish to have your name and/or address added, changed or deleted from our *ComputerNews* mailing list, please complete this form.

☐ **ADD**
☐ **CHANGE**
☐ **DELETE** **MY NAME AND ADDRESS.**

NAME, ADDRESS and TELEPHONE, in full

PREVIOUS NAME and ADDRESS, in full (or include your old mailing label)

☐ Check here if you can receive *ComputerNews* by campus mail.

Comments on this issue of *ComputerNews.*

———————————————— FOLD HERE ————————————————

Return to:

ComputerNews
University of Toronto Computing Services
4 Bancroft Avenue, Room 217
Toronto, Ontario
M5S 1A1